

Πανεπιστήμιο Πατρών
Τμήμα Διοίκησης Επιχειρήσεων
Μεταπτυχιακό Πρόγραμμα Σπουδών
«Νέες Αρχές στη Διοίκηση Επιχειρήσεων (MBA)»

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Θέμα: «Υποδείγματα μαθηματικού προγραμματισμού
για το σχεδιασμό του ωρολογίου προγράμματος ενός
εκπαιδευτικού ιδρύματος»



Επιβλέπων Καθηγητής: Ιωάννης Γιαννίκος

Φοιτήτρια: Δήμου Ελένη
Α.Μ.: 57

Πίνακας Περιεχομένων

Πίνακας Περιεχομένων	2
Περίληψη	5
1^ο Κεφάλαιο	6
Εισαγωγή στο χρονικό προγραμματισμό	6
1.1 Το ζήτημα του χρονικού προγραμματισμού	6
1.2 Το ζήτημα της Βελτιστοποίησης	8
1.3 Προσεγγίσεις επίλυσης του προβλήματος	9
1.4 Αυτόματος & ημιαυτόματος χρονικός προγραμματισμός	11
1.5 Συμπεράσματα	11
2^ο Κεφάλαιο	12
Σχεδιασμός ωρολόγιου προγράμματος Εκπαιδευτικών Ιδρυμάτων	12
2.1 Εισαγωγή	12
2.2 Βασικοί συντελεστές του προβλήματος	13
2.3 Κατηγοριοποίηση του προβλήματος	14
2.4 Οι περιορισμοί του προβλήματος	16
2.5 Στόχοι βελτιστοποίησης του προβλήματος	18
2.6 Ωρολόγιο πρόγραμμα μαθημάτων πανεπιστημίων	19
2.7 Μαθηματικό Μοντέλο Προβλήματος	19
2.8 Πρόβλημα Βελτιστοποίησης	21
2.9 Προσεγγίσεις επίλυσης και διαφορετικές διατυπώσεις του προβλήματος	22
2.9.1 Μέθοδοι στηριζόμενες στο μαθηματικό προγραμματισμό	22
2.9.2 Μέθοδοι επίλυσης με αναγωγή σε χρωματισμό γραφημάτων	22
2.9.3 Cluster μέθοδοι	22
2.9.4 Μέθοδοι επίλυσης βάσει των περιορισμών	23
2.9.5 Μετα-ευρετικές (Meta-heuristics) μέθοδοι	23
2.9.6 Προσέγγιση επίλυσης με βάση την πολυκριτηριακή ανάλυση	23
2.9.7 Case-based reasoning approaches	24
2.10 Συμπεράσματα	24
3^ο Κεφάλαιο	25
Μέθοδοι επίλυσης προβλημάτων χρονικού προγραμματισμού	25
3.1 Εισαγωγή	25

3.2 Ακέραιος γραμμικός προγραμματισμός.....	26
3.3 Εφαρμογή τεχνικών Ροής Δικτύων.....	28
3.4 Αναγωγή σε πρόβλημα Χρωματισμού Γραφήματος	29
3.5 Local Search based techniques	30
3.5.1 Hill Climbing	31
3.5.2. Simulated Annealing	31
3.5.3. The Threshold Acceptance Algorithm.....	32
3.5.4. Great Deluge Algorithm	33
3.5.5 Tabu Search	34
3.6 Εξελικτικοί αλγόριθμοι (Evolutionary algorithms)	35
3.6.1. Γενετικοί αλγόριθμοι (Genetic algorithms).....	35
3.6.2 Μιμητικοί αλγόριθμοι (Memetic algorithms)	37
3.6.3. Ant algorithms.....	37
3.7 Τεχνικές επίλυσης βάσει περιορισμών (Constraint based techniques).....	38
3.7.1. Constraint logic programming	38
3.8 Πολυκριτηριακή ανάλυση (Multi-Criteria techniques)	39
4^ο Κεφάλαιο	42
Πολυκριτηριακή ανάλυση βάσει συναινετικού προγραμματισμού	42
4.1 Εισαγωγή	42
4.2 Επιλογή μεθοδολογίας	43
4.3 Πολυκριτηριακή προσέγγιση βασιζόμενη στον συναινετικό προγραμματισμό (Multicriteria approach based on compromise programming)	44
4.3 Επίλυση 1 ^{ου} προβλήματος.....	51
4.3.1 Δεδομένα και απαιτήσεις του προβλήματος	51
4.3.2 Μοντελοποίηση του 1 ^{ου} προβλήματος.....	53
4.3.3 Επίλυση αντικειμενικής συνάρτησης Καθηγητών $\min(\sum d_j^+)$	55
4.3.4 Επίλυση αντικειμενικής συνάρτησης Τμημάτων $\min(\sum d_i^+)$	60
4.3.5 Ελαχιστοποίηση της απόστασης – Τελικό Ωρολόγιο Πρόγραμμα	65
4.4 Επίλυση 2 ^{ου} προβλήματος.....	71
4.4.1 Δεδομένα και απαιτήσεις του προβλήματος	71
4.4.2 Μοντελοποίηση 2 ^{ου} προβλήματος	73
4.4.3 Επίλυση αντικειμενικής συνάρτησης Καθηγητών $\min(\sum d_j^+)$	75

4.4.4 Επίλυση αντικειμενικής συνάρτησης Τμημάτων $\min(\sum d_i^+)$	82
4.4.5 Ελαχιστοποίηση της απόστασης – Τελικό Ωρολόγιο Πρόγραμμα	89
5ο Κεφάλαιο.....	97
Συμπεράσματα.....	97
Βιβλιογραφία	99
Παράρτημα	102
<i>Παράρτημα 1: Activity Template Data Collection Form</i>	<i>102</i>
<i>Παράρτημα 2: Παράδειγμα επίλυσης με Χρωματισμό Γραφήματος.....</i>	<i>105</i>
<i>Παράρτημα 3: Γενική μεθοδολογία μοντελοποίησης πολυκριτήριων γραμμικών προγραμμάτων.....</i>	<i>106</i>

Περίληψη

Η παρούσα διπλωματική εργασία πραγματεύεται το πρόβλημα του σχεδιασμού του ωρολογίου προγράμματος (timetable) μαθημάτων ενός εκπαιδευτικού ιδρύματος και πιο συγκεκριμένα ενός πανεπιστημίου. Λόγω της πολυπλοκότητας και της ύπαρξης πολυάριθμων μεταβλητών και παραμέτρων το πρόβλημα ανήκει στην κατηγορία των NP-complete προβλημάτων, γεγονός που κάνει την εξεύρεση της βέλτιστης λύσης μία πολύ δύσκολη υπόθεση.

Στο **1^ο Κεφάλαιο** γίνεται μία συνοπτική αναφορά στα προβλήματα χρονικού προγραμματισμού και στην έννοια της βελτιστοποίησης. Επίσης, επιγραμματικά παρατίθενται οι προσεγγίσεις επίλυσης που προτείνονται στη βιβλιογραφία

Στο **2^ο Κεφάλαιο** περιγράφεται το πρόβλημα του σχεδιασμού του ωρολογίου προγράμματος στα εκπαιδευτικά ιδρύματα, οι συντελεστές και οι στόχοι βελτιστοποίησης του προβλήματος καθώς και οι κατηγοριοποιήσεις ανάλογα με τον τύπο του ιδρύματος. Επίσης αναλύεται το μαθηματικό μοντέλο του προβλήματος, καθώς και η διατύπωση του ως πρόβλημα βελτιστοποίησης. Τέλος τίθενται ακροθιγώς οι διαφορετικές διατυπώσεις για την αντιμετώπιση του προβλήματος.

Στο **3^ο Κεφάλαιο** αναλύονται οι σημαντικότερες μέθοδοι και τεχνικές επίλυσης που χρησιμοποιούνται για την επίλυση αυτών των προβλημάτων. Οι τεχνικές προέρχονται από πολλούς και διαφορετικούς τομείς όπως για παράδειγμα το μαθηματικό προγραμματικό, τον ακέραιο γραμμικό προγραμματισμό, τις μετα-ευρετικές (meta-heuristics) μεθόδους, αλλά και από τον χώρο της Τεχνητής Νοημοσύνης.

Στο **4^ο Κεφάλαιο** επιλέγεται και περιγράφεται μία από τις μεθόδους επίλυσης, πιο συγκεκριμένα η πολυκριτηριακή ανάλυση βασιζόμενη στον συναινετικό προγραμματισμό (compromise programming). Στη συνέχεια του κεφαλαίου εφαρμόζεται η μέθοδος σε δεδομένα προβλήματος της σειράς Beasley.

Στο **5^ο Κεφάλαιο** καταγράφονται τα συμπεράσματα από τη θεωρητική ανασκόπηση αλλά και την πρακτική αξία της διαδικασίας της επίλυσης του προβλήματος του σχεδιασμού του ωρολογίου προγράμματος.

1^ο Κεφάλαιο

Εισαγωγή στο χρονικό προγραμματισμό

1.1 Το ζήτημα του χρονικού προγραμματισμού

Η έννοια του χρονικού προγραμματισμού περιγράφει την ανάγκη για το σχεδιασμό ενός ωρολογίου προγράμματος (timetable). Το 1996, ο Wren [18] περιγράφει το πρόβλημα του χρονικού προγραμματισμού, ως το πρόβλημα της αξιοποίησης συγκεκριμένων πόρων, λαμβάνοντας υπόψη την ύπαρξη κάποιων περιορισμών, σύμφωνα πάντα με περιορισμένο αριθμό χρονικών διαστημάτων, με σκοπό την ικανοποίηση αντικειμενικών στόχων στον υψηλότερο δυνατό βαθμό.

Τα προβλήματα του χρονικού προγραμματισμού εμφανίζουν υψηλή ποικιλομορφία και περικλείουν πολλούς και διαφορετικούς τομείς της ζωής μας. Ο ρόλος που διαδραματίζει ο ορθός σχεδιασμός του ωρολογίου προγράμματος (timetable) συχνά δεν γίνεται αντιληπτός, ή τουλάχιστον δεν είναι πάντα κατανοητός ο βαθμός στον οποίο ένα ωρολόγιο πρόγραμμα καθορίζει την καθημερινότητά μας. Τον υψηλό βαθμό σημαντικότητας των ωρολόγιων προγραμμάτων μπορούμε εύκολα να τον συμπεράνουμε, αν αναλογιστούμε την αξία τους και τη συμβολή στην ομαλή και αποδοτική λειτουργία και οργάνωση τομέων όπως η εκπαίδευση (ωρολόγια προγράμματα σχολείων-πανεπιστημίων), η υγεία (ωρολόγια προγράμματα νοσηλευτών και χειρουργείων), οι μεταφορές (προγράμματα τρένων, λεωφορείων κα.), αλλά και ο αθλητισμός (προγράμματα αγώνων).

Στην πλειονότητά τους τα προβλήματα του χρονικού προγραμματισμού είναι πολύπλοκα, συνδυαστικά προβλήματα και μάλιστα στη γενική μορφή τους είναι NP Complete. Η πολυπλοκότητα αυτή καθιστά πολύ δύσκολη, έως και αδύνατη, την αποδοτική και βέλτιστη επίλυσή τους.

Το γεγονός αυτό ανάγει την επίλυση τέτοιων προβλημάτων σε μία πολύ δύσκολη, χρονοβόρα, επίπονη και συχνά ακριβή διαδικασία. Για το λόγο αυτό, επιχειρούνται διαφορετικές διατυπώσεις του προβλήματος, ώστε να γίνει εφικτή η επίλυσή του. Σε κάποιες περιπτώσεις, η λύση ενός προβλήματος χρονικού προγραμματισμού ουσιαστικά επικεντρώνεται, απλώς και μόνο στην εύρεση ενός ωρολόγιου προγράμματος (timetable) που να ικανοποιεί όλους τους περιορισμούς. Σε αυτές τις περιπτώσεις το πρόβλημα διατυπώνεται ως **πρόβλημα αναζήτησης** (search problem). Σε κάποιες άλλες όμως περιπτώσεις το πρόβλημά μας διατυπώνεται ως **πρόβλημα βελτιστοποίησης** (optimization problem). Η απαίτηση που διατυπώνεται στο πρόβλημα βελτιστοποίησης είναι η εξεύρεση ενός ωρολόγιου προγράμματος που να ικανοποιεί όλους τους «σκληρούς» περιορισμούς (hard constraints) και να ελαχιστοποιεί (ή να μεγιστοποιεί) μία αντικειμενική συνάρτηση που εμπεριέχει τους «μαλακούς» περιορισμούς (soft constraints).

Αναφορικά με τις δύο προαναφερθείσες διατυπώσεις του προβλήματος έχει εκφραστεί και μία εναλλακτική προσέγγιση. Σύμφωνα με τους υποστηρικτές αυτής της προσέγγισης θεωρείται ότι η διατύπωση ενός προβλήματος βελτιστοποίησης στην πραγματικότητα, δεν είναι τίποτα άλλο, παρά η εφαρμογή τεχνικών βελτιστοποίησης σε ένα πρόβλημα αναζήτησης. Επίσης υποστηρίζουν ότι η αντικειμενική συνάρτηση στοχεύει στη μείωση του αριθμού των μη εφικτών λύσεων στο πρόβλημα, οπότε πρακτικά, αναπαριστά την απόσταση μίας λύσης από την εφικτή λύση του πραγματικού προβλήματος (**distance to feasibility**).

Και στις δύο περιπτώσεις (πρόβλημα αναζήτησης ή βελτιστοποίησης) ορίζουμε το θεμελιώδες πρόβλημα (**underlying problem**), που δεν είναι άλλο από το **πρόβλημα της επιλογής της λύσης**. Στα προβλήματα αναζήτησης η επιλογή γίνεται με βάση το εάν υπάρχει μία λύση, ενώ στα προβλήματα βελτιστοποίησης το πρόβλημα της επιλογής έγκειται στο εάν υπάρχει μία λύση που να αποδίδει δεδομένη αξία-τιμή στην αντικειμενική συνάρτηση.

Ήδη έχουμε αναφερθεί στην πολυπλοκότητα των προβλημάτων χρονικού προγραμματισμού. Στην ουσία όμως, αναφερόμαστε στην πολυπλοκότητα του θεμελιώδους προβλήματος της επιλογής. Το θεμελιώδες πρόβλημα ανήκει και αυτό στη κατηγορία των NP-Complete προβλημάτων σχεδόν για όλες τις παραλλαγές του προβλήματος. Για το λόγο αυτό, μία ακριβής λύση επιτυγχάνεται μόνο σε μικρού μεγέθους προβλήματα (π.χ. λιγότερα από 10 μαθήματα, για ένα ωρολόγιο πρόγραμμα πανεπιστημίου), αν και στην πραγματικότητα στα προβλήματα συνήθως εμπλέκονται

μερικές εκατοντάδες μαθημάτων. Έτσι είναι επόμενο πως μόνο ευρετικές μέθοδοι (heuristic methods) (Pearl, 1984) [3] είναι εφικτές και ενδείκνυται να εφαρμοστούν στη διαδικασία επίλυσης του προβλήματος, αν και στην πράξη, ούτε οι ευρετικές μέθοδοι εγγυώνται την εξεύρεση της βέλτιστης λύσης.

1.2 Το ζήτημα της Βελτιστοποίησης

Είναι αντιληπτό, από τη μέχρι τώρα αναφορά στα προβλήματα χρονικού προγραμματισμού, ότι το ζήτημα της βελτιστοποίησης αποτελεί κρίσιμο παράγοντα για την επίλυση και τη διατύπωσή τους. Η έννοια της βελτιστοποίησης θέτει ως στόχο του προβλήματος, να αναζητηθεί η βέλτιστη λύση ανάμεσα σε όλες τις πιθανές-εφικτές λύσεις. Με άλλα λόγια να βρεθεί μία λύση στην εφικτή περιοχή (feasible region), η οποία να δίνει την ελάχιστη (ή μέγιστη) τιμή-αξία στην αντικειμενική συνάρτηση.

Το ζήτημα της βελτιστοποίησης συναντάται πολύ συχνά σε προβλήματα που αντιμετωπίζουμε στην καθημερινότητά μας. Τα προβλήματα βελτιστοποίησης, αναγνωρίζονται κυρίως από το γεγονός ότι καλούμαστε να υπολογίσουμε και να επιλέξουμε την καλύτερη λύση και όχι απλά μία λύση. Παραδείγματα τέτοιων προβλημάτων είναι τα ακόλουθα:

- Σχεδίαση τηλεπικοινωνιακού δικτύου: Ποιο είναι το πιο αξιόπιστο;
- Προγραμματισμός παραγωγής: Ποιος δίνει την καλύτερη απόδοση;
- Δομή πρωτεϊνών στον τρισδιάστατο χώρο: Ποια έχει τη μικρότερη δυνατή ενέργεια;
- Ωρολόγιο πρόγραμμα σχολείου: Πως μπορούμε να ικανοποιήσουμε όλους τους περιορισμούς;

Μερικά από τα κλασσικά προβλήματα βελτιστοποίησης που ταλανίζουν ακόμα και σήμερα τους ερευνητές είναι: το πρόβλημα του περιπλανόμενου πωλητή (traveling salesman problem), το δίλημμα του κρατουμένου (prisoner's dilemma), το πρόβλημα των κιτιών συσκευασίας (bin packing problem), το πρόβλημα του σάκου

(knapsack problem), το cutting stock problem, το πρόβλημα του Κινέζου ταχυδρόμου (Chinese postman problem), και το πρόβλημα της δρομολόγησης οχημάτων (vehicle routing problem).

Οι μέθοδοι που προτείνονται γενικότερα για τα προβλήματα βελτιστοποίησης ποικίλλουν ανάλογα με τις ιδιαιτερότητες, τα χαρακτηριστικά και τους περιορισμούς του κάθε προβλήματος. Κάποιες από τις σημαντικότερες, που εμφανίζουν μεγάλη αποδοχή και χρησιμοποιούνται ευρέως από τους χρήστες είναι: ο δυναμικός προγραμματισμός, οι meta-heuristics (μετα-ευρετικές) μέθοδοι, η μέθοδος relaxation και η μέθοδος Simulated Annealing.

1.3 Προσεγγίσεις επίλυσης του προβλήματος

Δεδομένου ότι η ύπαρξη προβλημάτων χρονικού προγραμματισμού δεν είναι καθόλου ένα σύγχρονο ζήτημα, αλλά αντιθέτως προβληματίζει τους ερευνητές εδώ και δεκαετίες, είναι λογικό ότι αρχικά η προσέγγιση της λύσης γινόταν «με το χέρι», μία επίπονη διαδικασία που συχνά απαιτούσε την πολυήμερη εργασία αρκετών ατόμων. Επιπροσθέτως, οι παράμετροι που έπρεπε να ελεγχθούν ήταν πολλές, και συχνά ήταν αρκετά σύνθετες. Έγινε λοιπόν αντιληπτό, ότι με τη χειρωνακτική προσέγγιση της λύσης, ελλοχεύει πάντα ο κίνδυνος η λύση που θα προκύψει να μην ικανοποιεί όλες τις παραμέτρους του προβλήματος, δεδομένου ότι δεν υπάρχει συστηματική διαδικασία για να ελεγχθεί μία χειρωνακτική λύση ως προς την πληρότητά της. Για παράδειγμα, είναι πιθανόν να προκύψουν λύσεις όπου σύμφωνα με το ωρολόγιο πρόγραμμα μαθημάτων ενός πανεπιστημίου ένας φοιτητής δεν θα μπορεί να παρακολουθήσει τα μαθήματα που επιθυμεί γιατί ήταν προγραμματισμένα την ίδια ώρα.

Για τους παραπάνω λόγους, η προσοχή των ερευνητών εδώ και αρκετές δεκαετίες στράφηκε στον αυτοματοποιημένο χρονικό προγραμματισμό. Τα τελευταία σαράντα χρόνια πολλές μελέτες που σχετίζονται με τον αυτοματοποιημένο προγραμματισμό έχουν δημοσιευτεί, αρκετές εκ των οποίων έχουν εφαρμοσθεί με μεγάλη επιτυχία σε διάφορους οργανισμούς, ιδρύματα αλλά και επιχειρήσεις.

Είναι λοιπόν προφανές, ότι η πλειονότητα των πρώτων προσπαθειών αυτοματοποίησης της διαδικασίας που αναπτύχθηκαν για την επίλυση προβλημάτων χρονικού προγραμματισμού, βασίστηκαν στη μίμηση του χειρωνακτικού τρόπου επίλυσης του προβλήματος (υπολογισμός της λύσης «με το χέρι»). Όλες αυτές οι τεχνικές, που αποκαλούνται άμεσα ευρετικές μέθοδοι (**direct heuristics**) στηρίχθηκαν στη **διαδοχική επαύξηση του προβλήματος** (successive augmentation). Η μέθοδος αυτή χρησιμοποιούσε ένα ημιτελές ωρολόγιο πρόγραμμα και το επέκτεινε σταδιακά, προσθέτοντας ένα-ένα γεγονότα (π.χ. διάλεξη), ώσπου όλα τα γεγονότα (π.χ. οι διαλέξεις) να ενταχθούν στο πρόγραμμα και να κατασκευαστεί τελικά ένα πλήρες ωρολόγιο πρόγραμμα. Η βασική ιδέα της προσέγγισης αυτής ήταν ότι πρώτα έπρεπε να προγραμματιστεί η ώρα διεξαγωγής του γεγονότος που ήταν πιο δύσκολο να προγραμματιστεί (εμφάνιζε τους περισσότερους περιορισμούς). Η διαφορά των προσεγγίσεων αυτών κατά την εφαρμογή τους έγκειται στον ορισμό του τι σημαίνει ότι ένα γεγονός είναι δύσκολο να προγραμματιστεί.

Αργότερα, οι ερευνητές άρχισαν να εφαρμόζουν γενικές τεχνικές για την επίλυση των προβλημάτων χρονικού προγραμματισμού. Έτσι έκαναν την εμφάνισή τους προσεγγίσεις που βασίζονταν μεταξύ άλλων στον ακέραιο γραμμικό προγραμματισμό (**integer linear programming**), καθώς και στη ροή δικτύων (**network flow**). Στην εξέλιξη αυτή, αξίζει να σημειωθεί, ο ρόλος που διαδραμάτισε η αύξηση της ισχύος των υπολογιστών και η εξέλιξη της έρευνας στο χώρο των αλγορίθμων.

Επίσης, δεν ήταν λίγες οι περιπτώσεις που οι ερευνητές επιδίωξαν να διατυπώσουν σε μία διαφορετική βάση το πρόβλημα και να το ανάγουν σε πρόβλημα που μπορεί να επιλυθεί με μία μέθοδο, που είχε ήδη μελετηθεί αρκετά και ήταν περισσότερο κατανοητή, αυτή του χρωματισμού γραφημάτων (**graph coloring**).

Πρόσφατα, την εμφάνισή τους έκαναν με μεγάλη επιτυχία, κάποιες προσεγγίσεις που έκαναν χρήση τεχνικών που προέρχονται από το χώρο της Τεχνητής Νοημοσύνης (**Artificial Intelligence**). Μεταξύ άλλων ξεχωρίζουν οι τεχνικές: Simulated Annealing, Γενετικοί αλγόριθμοι, Tabu search και ικανοποίηση περιορισμών (constraint satisfaction).

1.4 Αυτόματος & ημιαυτόματος χρονικός προγραμματισμός

Πολλοί συγγραφείς πιστεύουν πως το πρόβλημα του χρονικού προγραμματισμού δεν μπορεί ή μάλλον δεν πρέπει να αυτοματοποιηθεί πλήρως. Το κύριο επιχείρημα που τους οδηγεί σε αυτή την πεποίθηση, είναι διττό: Από τη μία πλευρά υπάρχουν αιτίες που κάνουν ένα ωρολόγιο πρόγραμμα καλύτερο από ένα άλλο και οι οποίες δεν είναι καθόλου εύκολο να εκφραστούν σε ένα αυτόματο σύστημα. Από την άλλη πλευρά από τη στιγμή που το ερευνητικός χώρος για την εξεύρεση τη λύσης είναι συνήθως τεράστιος, η ανθρώπινη έρευνα θα μπορούσε να οδηγήσει το σύστημα επίλυσης σε ενδεχομένως καλύτερες λύσεις που ο υπολογιστής ίσως να μην μπορούσε μόνος του να τις εντοπίσει. Γι' αυτό υπάρχουν συστήματα που απαιτούν μεγάλο βαθμό συμμετοχής από τον άνθρωπο, επιτρέποντας στο χρήστη τουλάχιστον να ρυθμίζει χειροκίνητα την τελική εκροή (λύση) του συστήματος. Τα συστήματα αυτά καλούνται ημιαυτόματα (interactive), σε αντίθεση με τα υπόλοιπα που καλούνται αυτόματα (batch) συστήματα.

1.5 Συμπεράσματα

Αναγνωρίζοντας τη σημασία και τη συμβολή των ωρολόγιων προγραμμάτων (timetable) στην καθημερινότητά μας, μπορούμε να κατανοήσουμε την σημαντική ερευνητική δραστηριότητα που παρατηρείται στον τομέα του χρονικού προγραμματισμού. Ιδιαίτερος στον χώρο των εκπαιδευτικών ιδρυμάτων, οι προσεγγίσεις, οι μέθοδοι και οι τεχνικές επίλυσης των προβλημάτων αυτών είναι πολλές, προέρχονται από διαφορετικούς επιστημονικούς χώρους και συνεχώς εξελίσσονται όπως θα διαπιστώσουμε στα επόμενα κεφάλαια.

2^ο Κεφάλαιο

Σχεδιασμός ωρολόγιου προγράμματος Εκπαιδευτικών Ιδρυμάτων

2.1 Εισαγωγή

Το πρόβλημα του σχεδιασμού των ωρολόγιων προγραμμάτων (**Timetabling**) των εκπαιδευτικών ιδρυμάτων ανήκει στα προβλήματα χρονικού προγραμματισμού και έγκειται στον προγραμματισμό μίας ακολουθίας διαλέξεων-εργαστηρίων ή εξετάσεων για τους καθηγητές και φοιτητές του ιδρύματος σε μία καθορισμένη χρονική περίοδο, ικανοποιώντας μία πληθώρα περιορισμών. Με απλά λόγια αναφέρεται στον προσδιορισμό του εβδομαδιαίου ωρολογίου προγράμματος διδασκαλίας και εξετάσεων του εκπαιδευτικού ιδρύματος. Ίσως να μην φαντάζει τόσο σημαντικός, όσο στην πραγματικότητα είναι, ο ρόλος του εβδομαδιαίου προγράμματος διδασκαλίας ή εξετάσεων. Εάν όμως αναλογιστεί κανείς το σύνολο των ανθρώπων των οποίων καθορίζει τη διάθεση του χρόνου τους καθώς επίσης και τους πολύ συγκεκριμένους υλικούς πόρους που έχει να διαχειριστεί, θα καταλήξει μάλλον στο συμπέρασμα ότι αποτελεί ένα πολύ κομβικό παράγοντα για την ομαλή, παραγωγική και αποδοτική λειτουργία ενός εκπαιδευτικού ιδρύματος. Αποτελεί ίσως τη σημαντικότερη διοικητική και διαχειριστική λειτουργία των εκπαιδευτικών ιδρυμάτων. Για το λόγο αυτό έχει προσελκύσει τόσο μεγάλο ερευνητικό ενδιαφέρον, με ζητούμενο την εξεύρεση της βέλτιστης λύσης, του καλύτερου ωρολογίου προγράμματος.

2.2 Βασικοί συντελεστές του προβλήματος

Όπως είναι προφανές και από τον ορισμό του προβλήματος ωρολόγιου προγραμματισμού ενός εκπαιδευτικού ιδρύματος, οι βασικοί συντελεστές του είναι οι παρακάτω:

- το εκπαιδευτικό προσωπικό του ιδρύματος,
- οι φοιτητές,
- οι διαθέσιμες αίθουσες και
- οι ώρες διδασκαλίας (χρονικοί περίοδοι)

Όπως αναφέρθηκε ήδη στην εισαγωγή του κεφαλαίου, ο ανθρώπινος παράγοντας αποτελεί έναν από τους σημαντικότερους συντελεστές του προβλήματος. Εάν εξετάσουμε το ζήτημα του ωρολόγιου προγράμματος από την πλευρά του εκπαιδευτικού προσωπικού, δηλαδή των καθηγητών-διδασκόντων, μπορούμε εύκολα να διακρίνουμε την επιθυμία τους το πρόγραμμα να ανταποκρίνεται στις ανάγκες τους, οι οποίες βέβαια είναι συχνά διαφορετικές για τον καθένα ξεχωριστά. Οι προσδοκίες τους από το πρόγραμμα είναι πιθανόν να αφορούν σε κάποιες ελεύθερες μέρες μέσα στην εβδομάδα, ώστε να μπορούν να μετακινούνται είτε για τις επαγγελματικές τους υποχρεώσεις εντός και εκτός Ελλάδος, είτε για προσωπικούς λόγους. Επίσης, κάποιοι θα εκφράσουν ιδιαίτεροι προτίμηση για τις ώρες διδασκαλίας τους ή ακόμα και για τις αίθουσες διδασκαλίας συχνά λόγω της φύσης των μαθημάτων και του εξοπλισμού που απαιτείται.

Από την πλευρά των φοιτητών σίγουρα οι απαιτήσεις είναι αισθητά λιγότερες, όμως παρ' όλα αυτά σίγουρα θα εκφράσουν προτιμήσεις για τις ημέρες των μαθημάτων και ιδιαιτέρως των εξετάσεων, καθώς και για τις ώρες αφού συνήθως δεν επιθυμούν ούτε τα πολύ πρωινά ούτε τα βραδινά μαθήματα. Ένα πολύ κρίσιμο όμως ζήτημα που αφορά στους φοιτητές και καλείται να το σεβαστεί το ωρολόγιο πρόγραμμα είτε των μαθημάτων είτε των εξετάσεων, είναι αυτό το επικαλύψεων. Το ζήτημα αυτό, αφορά στις περιπτώσεις εκείνες που κάποιος φοιτητής πρέπει να παρακολουθήσει δύο μαθήματα τα οποία πραγματοποιούνται την ίδια ώρα και ημέρα. Τα φαινόμενα των επικαλύψεων πρέπει να αποκλείονται από το ωρολόγιο πρόγραμμα καθώς προκαλούν αδικίες ανάμεσα στους φοιτητές.

Αναφορικά με τις αίθουσες διδασκαλίας και εργαστηρίων εμφανίζεται συχνά πρόβλημα διαθεσιμότητας. Η έλλειψη διαθεσιμότητας είναι εμφανής κυρίως στις αίθουσες που φέρουν ειδικό εξοπλισμό, καθώς στην ελληνική πραγματικότητα, δεν είναι αρκετές ώστε να καλύπτουν τις ανάγκες καθηγητών και φοιτητών. Οποσδήποτε το ωρολόγιο πρόγραμμα πρέπει να τοποθετεί σωστά τα μαθήματα και τους καθηγητές στις κατάλληλες αίθουσες ώστε να τους δίνετε η δυνατότητα να είναι όσο αποδοτικοί μπορούν στη διδασκαλία τους.

Τέλος, το ωρολόγιο πρόγραμμα ενός εκπαιδευτικού ιδρύματος οφείλει να εξασφαλίζει τη σωστή κατανομή των διδακτικών ωρών των μαθημάτων στο εβδομαδιαίο πρόγραμμα και να ακολουθεί πάντα το πρόγραμμα σπουδών για τον καθορισμό των απαιτούμενων ωρών διαλέξεων και εργαστηρίων ανά εβδομάδα.

2.3 Κατηγοριοποίηση του προβλήματος

Πολλές διαφορετικές παραλλαγές προβλημάτων ωρολόγιου προγραμματισμού έχουν αναφερθεί στη βιβλιογραφία. Οι διαφορές που εμφανίζουν έγκεινται ουσιαστικά στους διαφορετικούς τύπους ιδρυμάτων που εξετάζουμε (πανεπιστήμιο ή σχολείο) και στον τύπο των περιορισμών που σε κάθε περίπτωση τίθενται. Μία γενική κατηγοριοποίηση παρουσιάζεται στη συνέχεια και προτείνει τρεις βασικές κατηγορίες προβλημάτων [1]:

Ωρολόγιος προγραμματισμός σχολείου (School Timetabling): Το εβδομαδιαίο πρόγραμμα για όλες τις τάξεις του σχολείου, αποφεύγοντας την ανάθεση ενός καθηγητή σε δύο τάξεις ταυτοχρόνως αλλά και την ανάθεση δύο καθηγητών σε μία τάξη την ίδια ώρα.

Ωρολόγιος προγραμματισμός μαθημάτων πανεπιστημίων (Course Timetabling): Το εβδομαδιαίο πρόγραμμα για όλες τις διαλέξεις και δραστηριότητες ενός μαθήματος, ελαχιστοποιώντας τις επικαλύψεις των διαλέξεων-δραστηριοτήτων που έχουν κοινούς φοιτητές.

Ωρολόγιος προγραμματισμός εξετάσεων (Examination Timetabling): Ο προγραμματισμός των εξετάσεων των μαθημάτων, αποφεύγοντας τις επικαλύψεις

των εξετάσεων των μαθημάτων που έχουν κοινούς φοιτητές, και προγραμματίζοντας ομοιόμορφα τις εξετάσεις στο διαθέσιμο χρονικό ορίζοντα.

Είναι σαφές ότι οι παραπάνω κατηγορίες εμφανίζουν διαφορετικά χαρακτηριστικά, καθώς απευθύνονται σε διαφορετικά εκπαιδευτικά ιδρύματα και κατά συνέπεια απαιτούν διαφορετικό χειρισμό για την επίλυσή τους. Το ίδιο ισχύει και για τους περιορισμούς του προβλήματος, οι οποίοι δεν είναι οι ίδιοι για τις τρεις κατηγορίες, αλλά διαφέρουν σημαντικά και έτσι επιβάλλουν την κατηγοριοποίηση των περιπτώσεων για την ορθότερη αντιμετώπιση του προβλήματος.

Φυσικά η ανωτέρω κατηγοριοποίηση δεν είναι κανόνας καθώς μπορεί να υπάρξουν κάποιες περιπτώσεις που βάσει των ιδιαίτερων χαρακτηριστικών τους μπορεί να βρίσκονται «ανάμεσα» σε δύο κατηγορίες. Για παράδειγμα ένα σχολείο το οποίο να παρέχει στους μαθητές μεγάλη ελευθερία κινήσεων σε ότι αφορά στην επιλογή παρακολούθησης των μαθημάτων τους τείνει να αντιμετωπιστεί ως ένα πρόβλημα ωρολόγιου προγραμματισμού διαλέξεων μαθημάτων και όχι σχολείου.

Επιπλέον, στη βιβλιογραφία έχουν αναλυθεί και άλλες κατηγοριοποίησης του προβλήματος. Για παράδειγμα, ο Bardadym [31] κατηγοριοποιεί το πρόβλημα του σχεδιασμού του ωρολογίου προγράμματος των πανεπιστημίων σε πέντε ομάδες:

Ωρολόγιο πρόγραμμα καθηγητών: Αναθέτει στους κατάλληλους καθηγητές τα μαθήματα.

Ωρολόγιο πρόγραμμα τμημάτων-καθηγητών: Καθορίζει τα μαθήματα σύμφωνα με το πρόγραμμα των τμημάτων των φοιτητών.

Ωρολόγιο πρόγραμμα μαθημάτων: Αναθέτει τα μαθήματα σύμφωνα με το πρόγραμμα του κάθε μεμονωμένου φοιτητή.

Ωρολόγιο πρόγραμμα εξετάσεων: Αναθέτει τις εξετάσεις στους φοιτητές, με τρόπο ώστε οι φοιτητές να μην δίνουν δύο εξετάσεις την ίδια ώρα.

Ανάθεση αιθουσών: Έπειτα από τον καθορισμό των τμημάτων-καθηγητών, τα αναθέτει σε συγκεκριμένες αίθουσες.

Βέβαια, ούτε η κατηγοριοποίηση σύμφωνα με τον Bardadym αποτελεί κανόνα, καθώς το κάθε εκπαιδευτικό ίδρυμα χαρακτηρίζεται από τις δικές του ανάγκες και ιδιαιτερότητες.

2.4 Οι περιορισμοί του προβλήματος

Έχει ήδη γίνει μνεία ότι στα προβλήματα βελτιστοποίησης σημαντικό ρόλο για τον προσδιορισμό των λύσεων διαδραματίζουν οι περιορισμοί που τίθενται στο πρόβλημα και ο βαθμός στον οποίο είναι απαραίτητο να ικανοποιείται ο καθένας από αυτούς. Δεν αποτελεί εξαίρεση το πρόβλημα του ωρολόγιου προγραμματισμού ενός εκπαιδευτικού ιδρύματος γι' αυτό και οι λύσεις οφείλουν να ικανοποιούν ένα σύνολο περιορισμών.

Οι περιορισμοί που τίθενται στα προβλήματα βελτιστοποίησης διακρίνονται σε «σκληρούς» (hard) και «μαλακούς» (soft) περιορισμούς. Η κατηγοριοποίηση αυτή έγκειται στο κατά πόσο ένας περιορισμός επιτρέπεται ή όχι να παραβιαστεί στη λύση που προτείνεται. Πιο συγκεκριμένα υπάρχουν οι «**σκληροί**» **περιορισμοί**, οι οποίοι αποτελούν αυστηρές και συγκεκριμένες απαιτήσεις, που δεν επιτρέπεται σε καμία περίπτωση να παραβιαστούν κατά την επίλυση του προβλήματος. Οι περιορισμοί συγκεκριμένα για τον ωρολόγιο προγραμματισμό των εκπαιδευτικών ιδρυμάτων εκφράζουν κυρίως, δύο συγκεκριμένες απαιτήσεις:

- Κανένας άνθρωπος δεν μπορεί να βρίσκεται σε δύο μέρη την ίδια χρονική στιγμή. Όπως είναι λογικό κάτι τέτοιο υποδηλώνει ότι κανένας φοιτητής δεν μπορεί να παρακολουθεί δύο διαλέξεις ταυτοχρόνως, και κανένας καθηγητής δεν μπορεί να διδάσκει σε δύο διαφορετικές αίθουσες την ίδια χρονική περίοδο. Η απαίτηση αυτή μας οδηγεί στον εξής περιορισμό όσον αφορά στην επίλυση του προβλήματός μας: ότι δύο μαθήματα που παρακολουθούνται από κοινούς φοιτητές, και τα μαθήματα που διδάσκονται από τον ίδιο καθηγητή δεν επιτρέπεται να τοποθετηθούν την ίδια χρονική περίοδο στο ωρολόγιο πρόγραμμα.
- Κάθε διάλεξη λαμβάνει χώρα σε μία συγκεκριμένη αίθουσα. Η απαίτηση αυτή μας υπαγορεύει την ανάγκη να υπάρχουν διαθέσιμες αίθουσες που να καλύπτουν τις ανάγκες των φοιτητών για όλα τα μαθήματα που τοποθετούνται την ίδια χρονική στιγμή στο ωρολόγιο πρόγραμμα.

Υπάρχουν όμως και οι «μαλακοί» περιορισμοί. Οι περιορισμοί αυτοί εκφράζουν επιθυμητά και όχι απόλυτα κριτήρια γεγονός που συνεπάγεται ότι είναι εφικτό να παραβιαστούν, φυσικά στη μικρότερη δυνατή κλίμακα. Η περιγραφή των «μαλακών» περιορισμών διαφέρει για κάθε εκπαιδευτικό ίδρυμα. Οι «μαλακοί» περιορισμοί μπορούν να αποκαλεστούν και **ποιοτικοί κανόνες**, μιας και το κατά πόσο παραβιάζονται αναπαρίσταται στο κατά πόσο ελαχιστοποιείται/μεγιστοποιείται η τιμή της αντικειμενικής συνάρτησης του προβλήματος. Κάποιοι από τους «μαλακούς» περιορισμούς είναι οι εξής:

- προτιμήσεις των καθηγητών
- εξάλειψη κενών ωρών στα προγράμματα εργασίας των καθηγητών ή των φοιτητών
- Χρονική απόσταση πολλαπλών ωρών του ίδιου τμήματος
- Ελάχιστος χρόνος μεταφοράς καθηγητών
- Ελάχιστος χρόνος μεταφοράς τμημάτων του ίδιου εξαμήνου

Οι μαλακοί περιορισμοί που αναφέρονται είναι ενδεικτικοί καθώς, όπως αναφέραμε, στην πραγματικότητα είναι πάρα πολλοί, είναι αδύνατο να ικανοποιηθούν όλοι και διαφέρουν σημαντικά, ανάλογα με το ίδρυμα που κάθε φορά εξετάζουμε.

Για την ορθότερη και πλήρη καταγραφή, αποτύπωση των προτιμήσεων των καθηγητών, ώστε να γίνει αποτελεσματικότερη η διδασκαλία των μαθημάτων, αλλά και για τον προσδιορισμό των ιδιαιτεροτήτων και των απαιτήσεων του κάθε μαθήματος, κάποια εκπαιδευτικά ιδρύματα έχουν δημιουργήσει συγκεκριμένες φόρμες συμπλήρωσης για τη συλλογή αυτών των δεδομένων και την «έκφρασή» τους στο ωρολόγιο πρόγραμμα. Ο σκοπός αυτής της λεπτομερούς καταγραφής είναι ο σαφέστερος προσδιορισμός των μαλακών περιορισμών, για να μπορέσουμε να οδηγηθούμε σε ένα ωρολόγιο πρόγραμμα που να ικανοποιεί τους ποιοτικούς κανόνες.

Στο Παράρτημα παρατίθεται η φόρμα συμπλήρωσης για τη συλλογή δεδομένων που χρησιμοποιείται στο Πανεπιστήμιο της Μελβούρνης (Παράρτημα 1, Πηγή: <http://www.unimelb.edu.au>).

2.5 Στόχοι βελτιστοποίησης του προβλήματος

Η πρωταρχική επιδίωξη κατά το σχεδιασμό ενός ωρολογίου προγράμματος είναι να ικανοποιούνται όλοι οι «σκληροί» περιορισμοί και να παραβιάζονται όσο το δυνατό λιγότεροι «μαλακοί» περιορισμοί. Έτσι διαμορφώνονται κάποιοι στόχοι που η επίτευξη τους εξασφαλίζει και καθορίζει το επίπεδο βελτιστοποίησης της λύσης του προβλήματος.

Δεν είναι δύσκολο να φανταστούμε σε τι αφορούν οι στόχοι βελτιστοποίησης ενός προβλήματος που άπτεται του σχεδιασμού του ωρολογίου προγράμματος ενός εκπαιδευτικού ιδρύματος. Οι στόχοι επικεντρώνονται στην ομαλή και αποτελεσματική λειτουργία του προγράμματος, στο βαθμό ικανοποίησης των καθηγητών για το συγκεκριμένο πρόγραμμα, τη σωστή αξιοποίηση των υλικών πόρων του ιδρύματος και τέλος τη θετική αποδοχή του προγράμματος από τους φοιτητές. Μερικοί από τους στόχους βελτιστοποίησης παρατίθενται πιο κάτω:

- Ελαχιστοποίηση κενών ωρών καθηγητών
- Ισοκατανομή ωρών διδασκαλίας ανά καθηγητή
- Ελαχιστοποίηση χρόνου μεταφοράς καθηγητών
- Ελαχιστοποίηση χρόνου μεταφοράς τμημάτων του ίδιου εξαμήνου
- Πολλαπλές ώρες του ίδιου τμήματος να πραγματοποιούνται στην ίδια αίθουσα
- Πολλαπλές ώρες του ίδιου τμήματος να ανατίθενται στον ίδιο καθηγητή
- Ελαχιστοποίηση των συνεχόμενων ωρών εξετάσεων
- Ελαχιστοποίηση του σχεδιασμού εξετάσεων την ίδια μέρα

Είναι προφανές από τη φύση του προβλήματος ότι είναι σχεδόν αδύνατο να ικανοποιούνται στον ίδιο βαθμό (μέγιστο) όλοι οι στόχοι βελτιστοποίησης του προβλήματος. Για το λόγο αυτό επιδιώκουμε τη μέγιστη ικανοποίηση όσο το δυνατόν περισσότερων στόχων.

2.6 Ωρολόγιο πρόγραμμα μαθημάτων πανεπιστημίων

Είναι σαφές, ότι ο σχεδιασμός του ωρολογίου προγράμματος μαθημάτων είτε αυτό αφορά σε σχολείο, είτε σε πανεπιστήμιο, συνίσταται στον προγραμματισμό των διαλέξεων και εργαστηρίων του κάθε μαθήματος σύμφωνα με ένα καθορισμένο αριθμό αιθουσών και χρονικών περιόδων, συνήθως σε εβδομαδιαία βάση.

Η βασική διαφορά του προβλήματος του ωρολογίου προγραμματισμού των σχολείων με αυτό ενός πανεπιστημίου, έγκειται στο γεγονός ότι τα μαθήματα του πανεπιστημίου μπορεί να έχουν κοινούς φοιτητές, ενώ στο σχολείο υπάρχουν συγκεκριμένα τμήματα μαθητών. Εάν δύο μαθήματα έχουν κοινούς φοιτητές δεν πρέπει να προγραμματισθούν στην ίδια περίοδο, είναι πολύ κρίσιμη η αποφυγή συγκρούσεων-επικαλύψεων μεταξύ μαθημάτων που μπορεί να παρακολουθεί μία ομάδα φοιτητών. Ακόμη, το πρόγραμμα μαθημάτων ενός σχολείου, αποτελείται από χρονικές περιόδους ίδιας διάρκειας και επαναλαμβάνεται σε εβδομαδιαία βάση όλο το σχολικό έτος.

Επιπροσθέτως, σε ένα πανεπιστήμιο, η διαθεσιμότητα των αιθουσών διαδραματίζει ένα πολύ σημαντικό ρόλο, ενώ στα σχολεία έχει ελάχιστη σημασία, αφού σχεδόν πάντα η κάθε τάξη έχει τη δική της αίθουσα. Για τα πανεπιστήμια μάλιστα, η έλλειψη διαθεσιμότητας αιθουσών και εξοπλισμού αναγκάζει το ωρολόγιο πρόγραμμα να υπακούει σε ένα σύνολο από περιορισμούς οι οποίοι δεν υπάρχουν όταν σχεδιάζουμε το πρόγραμμα ενός σχολείου.

Στην παρούσα εργασία έχει επιλεγεί να μελετηθεί το πρόβλημα του σχεδιασμού του ωρολογίου προγράμματος μαθημάτων, όπως αυτό αναδεικνύεται στο περιβάλλον ενός πανεπιστημίου.

2.7 Μαθηματικό Μοντέλο Προβλήματος

Εάν και υπάρχουν ποικίλες μορφοποιήσεις του προβλήματος σχεδιασμού του ωρολογίου προγράμματος μαθημάτων (Tripathy, 1992) [8], έχει επιλεγεί και παρουσιάζεται η μορφοποίηση σύμφωνα με τον De Werra, 1985 [5].

Έστω ότι υπάρχουν q μαθήματα $K_1, \dots, K_q$, και για κάθε i μάθημα K_i υπάρχουν k_i διαλέξεις. Υπάρχουν r ομάδες μαθημάτων $S_1, \dots, S_r$, που αντιπροσωπεύουν ομάδες μαθημάτων που έχουν κοινούς φοιτητές. Γεγονός που σημαίνει ότι τα μαθήματα που ανήκουν στο S_1 πρέπει όλα να προγραμματιστούν διαφορετικές ώρες. Ο αριθμός των χρονικών περιόδων είναι p και l_k ο μέγιστος αριθμός διαλέξεων που μπορούν να προγραμματιστούν στην περίοδο k (δηλαδή ο αριθμός των διαθέσιμων αιθουσών στην περίοδο k). Με βάση τα παραπάνω το πρόβλημά μας μπορεί να οριστεί ως εξής:

Να βρεθεί η τιμή της μεταβλητής: y_{ik} ($i = 1 \dots q$; $k = 1 \dots p$)

με τους περιορισμούς
(μ.τ.σ)

$$\sum_{k=1}^p y_{ik} = k_i \quad (i = 1 \dots q) \quad (1)$$

$$\sum_{i=1}^q y_{ik} \leq l_k \quad (k = 1 \dots p) \quad (2)$$

$$\sum_{i \in S_l} y_{ik} \leq 1 \quad (l = 1 \dots r ; k = 1 \dots p) \quad (3)$$

$$y_{ik} = 0 \text{ or } 1 \quad (i = 1 \dots q ; k = 1 \dots p) \quad (4)$$

Όπου $y_{ik} = 1$ εάν η διάλεξη ενός μαθήματος K_i προγραμματίστηκε στην περίοδο k , διαφορετικά $y_{ik} = 0$.

Ο περιορισμός (1) απαιτεί ότι κάθε μάθημα αποτελείται από το σωστό αριθμό διαλέξεων. Ο περιορισμός (2) εξασφαλίζει ότι την ίδια ώρα δεν υπάρχουν περισσότερες διαλέξεις από τις αίθουσες. Ο περιορισμός (3) εξασφαλίζει την αποφυγή επικαλύψεων-συγκρούσεων των διαλέξεων μεταξύ της ίδιας ομάδας μαθημάτων. Το παραπάνω μοντέλο αποδεικνύεται ότι είναι NP-complete αν αναχθεί σε πρόβλημα χρωματισμού γραφημάτων (graph coloring problem).

Αξίζει να σημειωθεί ότι υπάρχει μία ισοδύναμη διατύπωση του προβλήματος όπου μπορούμε να εισαγάγουμε τη χρήση ενός πίνακα συγκρούσεων (conflicts matrix) αντί των ομάδων μαθημάτων που χρησιμοποιήθηκαν στο παραπάνω

υπόδειγμα, ώστε να αποφύγουμε τις συγκρούσεις-επικαλύψεις διαλέξεων με κοινούς φοιτητές. Ο πίνακας συγκρούσεων $C_{q \times q}$ είναι ένας δυαδικός πίνακας (binary matrix) όπου $c_{ij} = 1$ εάν το μάθημα K_i και K_j έχουν κοινούς φοιτητές (άρα δεν μπορούν να προγραμματιστούν την ίδια χρονική περίοδο), διαφορετικά $c_{ij} = 0$.

2.8 Πρόβλημα Βελτιστοποίησης

Στο παραπάνω μαθηματικό υπόδειγμα του προβλήματος ωρολόγιου προγραμματισμού είναι σκόπιμο να εισαγάγουμε την έννοια της βελτιστοποίησης. Η αντικειμενική συνάρτηση, σύμφωνα με τον De Werra, 1985 [5] του προβλήματος είναι η παρακάτω:

$$\max \sum_{i=1}^q \sum_{k=1}^p d_{ik} y_{ik}$$

Όπου d_{ik} αναπαριστά το πόσο επιθυμητή είναι διδασκαλία του μαθήματος K_i την χρονική περίοδο k .

Ο Tripathy, 1992 [8] προτείνει την δημιουργία ενός πίνακα συγκρούσεων (conflict matrix) $C_{q \times q}$ για την επίλυση του προβλήματος, με ακέραιες τιμές, έτσι ώστε το c_{ij} αντιπροσωπεύει τον αριθμό των κοινών φοιτητών (των φοιτητών που παρακολουθούν και τα δύο μαθήματα) των μαθημάτων K_i και K_j . Ο στόχος της βελτιστοποίησης για αυτή την προσέγγιση είναι η μείωση του αριθμού των φοιτητών, άρα και της δυσaréσκειας αποδοχής του προγράμματος, που έχουν μαθήματα τα οποία διδάσκονται την ίδια χρονική περίοδο.

Εκτός από αυτή τη βασική μορφή του προβλήματος που περιγράψαμε έχουν παρουσιαστεί και διάφορες επεκτάσεις αυτού. Σημειώνουμε ότι αφού το πρόβλημα στην απλή μορφή του είναι NP-complete το ίδιο ισχύει και για τις επεκτάσεις του. Παραδείγματα επεκτάσεων είναι η εισαγωγή προ-αναθέσεων στις διαλέξεις, η εισαγωγή της διαθεσιμότητας ανάθεσης των μαθημάτων καθώς επίσης και η εισαγωγή διαλέξεων μεταβλητής διάρκειας.

2.9 Προσεγγίσεις επίλυσης και διαφορετικές διατυπώσεις του προβλήματος

Πληθώρα προσεγγίσεων, διατυπώσεων και τεχνικών επίλυσης του προβλήματος σχεδιασμού του ωρολογίου προγράμματος ενός πανεπιστημίου περιγράφονται στη βιβλιογραφία και οι περισσότερες έχουν δοκιμαστεί από πολλά πανεπιστήμια στην πράξη. Οι προσεγγίσεις επίλυσης θα μπορούσαν να κατηγοριοποιηθούν σύμφωνα με το βαθμό του πληρούν συγκεκριμένα κριτήρια. Στη συνέχεια παρουσιάζεται η κατηγοριοποίηση που προτείνουν οι Carter και Laporte (1996 και 1998) [19-21] που βασίζεται στις διαφορετικές μεθοδολογίες που χρησιμοποιούνται για την επίλυση αυτών των προβλημάτων.

2.9.1 Μέθοδοι στηριζόμενες στο μαθηματικό προγραμματισμό

Σύμφωνα με αυτή την προσέγγιση το πρόβλημά μας διαμορφώνεται σε πρόβλημα γραμμικού ακέραιου προγραμματισμού και παρουσιάζεται από τον Akkoyunly (1973) [22]. Οι υψηλές υπολογιστικές απαιτήσεις αυτών των μεθόδων τις καθιστά όχι και τόσο ελκυστικές, κυρίως για μεγάλα προβλήματα σχεδιασμού ωρολόγιων προγραμμάτων.

2.9.2 Μέθοδοι επίλυσης με αναγωγή σε χρωματισμό γραφημάτων

Το πρόβλημα του χρονικού προγραμματισμού πανεπιστημίων μπορεί να παρουσιαστεί και να αναχθεί σε πρόβλημα χρωματισμού γραφήματος.

2.9.3 Cluster μέθοδοι

Σύμφωνα με αυτές τις μεθόδους, αρχικά χωρίζουμε σύνολα «γεγονότων» σε ομάδες έτσι ώστε τα γεγονότα που βρίσκονται στην ίδια ομάδα να ικανοποιούν τους σκληρούς περιορισμούς. Στη συνέχεια οι ομάδες ανατίθενται σε χρονικές περιόδους έτσι ώστε να εκπληρώνονται οι μαλακοί περιορισμοί. Όμως, οι μέθοδοι αυτές δίνουν χαμηλής ποιότητας αποτελέσματα.

2.9.4 Μέθοδοι επίλυσης βάσει των περιορισμών

Στις μεθόδους που βασίζονται στους περιορισμούς τα γεγονότα του προβλήματος εκφράζονται με ένα σύνολο μεταβλητών των οποίων οι τιμές (π.χ. υλικοί πόροι όπως οι αίθουσες και οι χρονικές περίοδοι) προσδιορίζονται σύμφωνα με τον αριθμό των περιορισμών. Όταν η διασπορά των τιμών που προσδιορίσαμε οδηγεί σε μία μη εφικτή λύση επαναλαμβάνουμε τη διαδικασία έως ότου, επαναπροσδιορίζοντας τις τιμές των μεταβλητών, προκύψει μία λύση που να ικανοποιεί όλους τους περιορισμούς.

2.9.5 Μετα-ευρετικές (Meta-heuristics) μέθοδοι

Τις τελευταίες δύο δεκαετίες έχει αυξηθεί το ερευνητικό ενδιαφέρον για την ανάπτυξη των μετα-ευρετικών (meta-heuristics) μεθόδων για την επίλυση του χρονικού προγραμματισμού στα πανεπιστήμια. Στις μετα-ευρετικές μεθόδους περιλαμβάνονται διάφορες προσεγγίσεις-τεχνικές επίλυσης όπως Simulated Annealing, Tabu Search, Γενετικοί αλγόριθμοι κα. Πιο συγκεκριμένα χωρίζονται σε δύο υποκατηγορίες:

- **Local Search based techniques**
- **Εξελικτικοί αλγόριθμοι (Evolutionary algorithms)**

Οι μέθοδοι αυτές ξεκινούν με μία ή περισσότερες αρχικές λύσεις και εφαρμόζοντας ερευνητικές στρατηγικές προσεγγίζουν το τοπικό βέλτιστο. Το πλεονέκτημα των ερευνητικών αλγορίθμων είναι ότι μας παρέχουν υψηλές ποιοτικές λύσεις. Ωστόσο, αρκετά συχνά, απαιτούν σημαντικό συντονισμό των παραμέτρων και επίσης εμφανίζουν αξιοσημείωτο υπολογιστικό κόστος.

Σε επόμενο κεφάλαιο θα αναλυθεί ποιες τεχνικές περιλαμβάνονται στις Local Search techniques και ποιοι είναι οι εξελικτικοί αλγόριθμοι.

2.9.6 Προσέγγιση επίλυσης με βάση την πολυκριτηριακή ανάλυση

Η επίλυση που στηρίζεται στην πολυκριτηριακή ανάλυση, υπολογίζει το βαθμό στον οποίο ένα κριτήριο παραβιάζει τον αντίστοιχο περιορισμό. Τα κριτήρια εμφανίζουν διαφορετικά επίπεδα σημαντικότητας, σε διαφορετικές καταστάσεις και σε διαφορετικά ιδρύματα. Η πολυκριτηριακή προσέγγιση επίλυσης που βασίζεται

στον Compromise Programming και η οποία μπορεί να διαχειριστεί όλα τα κριτήρια ταυτόχρονα έχει περιγραφεί από τους Bruke et al. (2001a) [23]. Υπάρχει επίσης μία πολυκριτηριακή «επέκταση» του Great Deluge Algorithm η οποία οδηγεί την ερευνητική διαδικασία του ερευνητικού χώρου από την δυναμική αλλαγή της στάθμησης των κριτηρίων παρουσιάστηκε από τους Petrovic και Bykov (2003) [24].

2.9.7 Case-based reasoning approaches

Οι μέθοδοι αυτές αναπτύχθηκαν σχετικά πρόσφατα. Χρησιμοποιούν παλιές λύσεις ως θεμέλιους λίθους για να κατασκευάσουν μία λύση που να οδηγήσει σε ένα νέο ωρολόγιο πρόγραμμα (Burke et al., 2003b), (Burke et al., 2002a), (Burke et al., 2002b), (Burke et al., 2001c), (Burke et al., 2001d) και (Petrovic and Qu, 2002) [25-30]. Το ζήτημα του καθορισμού τρόπου μέτρησης της ομοιότητας μεταξύ δύο ωρολογίων προγραμμάτων παίζει έναν κρίσιμο ρόλο στις μεθόδους αυτές.

2.10 Συμπεράσματα

Είναι κατανοητό, μετά και την περιγραφή των συντελεστών, των στόχων βελτιστοποίησης αλλά και τις κατηγοριοποιήσεις του προβλήματος ότι πρόκειται για ένα σύνθετο και πολύπλευρο πρόβλημα, του οποίου η αποτελεσματική επίλυση δεν είναι καθόλου εύκολη. Για το λόγο αυτό έχουν αναπτυχθεί πολλές και διαφορετικές διατυπώσεις του προβλήματος που βασίζονται σε διαφορετικές προσεγγίσεις και τεχνικές επίλυσης. Κρίνεται λοιπόν σκόπιμο, στη συνέχεια της παρούσας εργασίας, να αναλυθούν οι σημαντικότερες και περισσότερο χρησιμοποιούμενες μεθοδολογίες και τεχνικές επίλυσης, ώστε να καταγραφεί μία ολοκληρωμένη θεώρηση της αντιμετώπισης του προβλήματος σχεδιασμού ωρολογίου προβλήματος μαθημάτων ενός εκπαιδευτικού ιδρύματος.

3^ο Κεφάλαιο

Μέθοδοι επίλυσης προβλημάτων χρονικού προγραμματισμού

3.1 Εισαγωγή

Όπως αναφέρθηκε και στην παράγραφο 2.9 του 2^{ου} Κεφαλαίου, οι προσεγγίσεις, μεθοδολογίες και τεχνικές που χρησιμοποιούνται και προτείνονται στη βιβλιογραφία για το σχεδιασμό του ωρολογίου προγράμματος ενός εκπαιδευτικού ιδρύματος και γενικότερα για την επίλυση ενός προβλήματος χρονικού προγραμματισμού είναι ποικίλες και με διαφορετικές προελεύσεις.

Στο κεφάλαιο αυτό επιχειρείται η παράθεση των κατευθυντήριων γραμμών της κάθε μεθόδου επίλυσης ώστε να γίνουν σαφείς οι διαφορές, οι ομοιότητες, τα πλεονεκτήματα και οι δυσκολίες που θα αντιμετωπίσει ο χρήστης κατά την επίλυση ενός προβλήματος χρονικού προγραμματισμού. Οι προσεγγίσεις-τεχνικές που θα μελετηθούν είναι οι ακόλουθες:

- ❖ Ακέραιος γραμμικός προγραμματισμός (Integer Linear Programming)
- ❖ Ροή δικτύων (Network Flow)
- ❖ Χρωματισμός γραφήματος (Graph Coloring)
- ❖ Local Search based techniques
 - Hill Climbing
 - Simulated Annealing
 - The Threshold Acceptance Algorithm
 - Great Deluge Algorithm
 - Tabu Search
- ❖ Εξελικτικοί αλγόριθμοι (Evolutionary algorithms)

- Γενετικοί αλγόριθμοι
- Μιμητικοί αλγόριθμοι
- Ant αλγόριθμοι
- ❖ Τεχνικές επίλυσης βάσει περιορισμών (Constraint based techniques)
 - Constraint logic programming techniques
- ❖ Πολυκριτηριακή ανάλυση (Multi-Criteria techniques)

Ίσως στη βιβλιογραφία αναφέρονται και προτείνονται κάποιες ακόμα μέθοδοι επίλυσης, η επιλογή όμως που έχει πραγματοποιηθεί περιλαμβάνει τις πιο διαδεδομένες και χρησιμοποιούμενες μεθόδους από τα εκπαιδευτικά ιδρύματα.

3.2 Ακέραιος γραμμικός προγραμματισμός

Το πρόβλημα του σχεδιασμού του ωρολογίου προγράμματος των μαθημάτων των πανεπιστημίων τις περισσότερες φορές αντιμετωπίζεται με τη χρήση του ακέραιου γραμμικού προγραμματισμού (**Integer Linear Programming**).

Ο γραμμικός προγραμματισμός (Linear Programming) αποτελεί αναμφίβολα το δημοφιλέστερο εργαλείο επίλυσης στο χώρο της επιχειρησιακής έρευνας και της διοικητικής επιστήμης (management science). Ο ακέραιος προγραμματισμός αποτελεί ένα ισχυρό εργαλείο κατασκευής **μαθηματικών μοντέλων** για προβλήματα χρονικού προγραμματισμού καθώς επιτρέπει την έκφραση των σχέσεων μεταξύ των εμπλεκόμενων οντοτήτων του προβλήματος με τον ορισμό των αντίστοιχων μεταβλητών καθώς και την έκφραση των κανόνων του προβλήματος με την εισαγωγή αντίστοιχων περιορισμών. Με τη χρήση των αντικειμενικών συναρτήσεων εκφράζονται αποδοτικά και με συστηματικό τρόπο τα ποιοτικά χαρακτηριστικά («μαλακοί» περιορισμοί) των λύσεων του προβλήματος με απώτερο στόχο την εύρεση της βέλτιστης ή, στις περιπτώσεις που κάτι τέτοιο δεν είναι εφικτό, μιας πολύ καλής λύσης.

Το σπουδαιότερο εμπόδιο που υπάρχει κατά την επίλυση ενός προβλήματος χρονικού προγραμματισμού δεν είναι άλλο από την πολυπλοκότητα της φύσης του προβλήματος που συνεπάγεται η ύπαρξη πολλών μεταβλητών.

Έχει ήδη αναφερθεί και σε προηγούμενα κεφάλαια ότι έχουμε να κάνουμε με ένα πρόβλημα που ανήκει στην κατηγορία των NP-Complete προβλημάτων. Το γεγονός αυτό, οδηγεί στην ανάπτυξη μαθηματικών μοντέλων με τεράστιο αριθμό μεταβλητών. Η συνέπεια του γεγονότος αυτού είναι δυστυχώς να καθίσταται μη εφικτή η επίλυση τους, είτε γιατί είναι αδύνατο να δημιουργηθούν όλες οι μεταβλητές του προβλήματος είτε λόγω αδυναμίας των εργαλείων επίλυσης προβλημάτων ακέραιου προγραμματισμού για επίλυση πολύ μεγάλων προβλημάτων.

Έτσι, ήταν αναπόφευκτο να ξεκινήσει στην ερευνητική κοινότητα μία προσπάθεια για εύρεση μεθόδων οι οποίες θα έλυναν μεγάλα προβλήματα γραμμικού και ακέραιου προγραμματισμού χωρίς να απαιτείται η εκ των προτέρων κατασκευή όλων των μεταβλητών του προβλήματος. Οι πρώτες επιτυχείς προσπάθειες της φιλοσοφίας αυτής εμφανίστηκαν πριν από πολλές δεκαετίες όταν προσπάθησαν οι Ford και Fulkerson να λύσουν ένα πρόβλημα multi-commodity flow, χωρίς όμως να λαμβάνουν υπόψη τους σε κάθε βήμα όλες τις μεταβλητές του προβλήματος, αλλά μόνο ένα μέρος αυτών. Αυτή την ιδέα υιοθέτησαν και βελτίωσαν οι Dantzig και Wolfe στο κάνοντας την πιο γενική για προβλήματα γραμμικού προγραμματισμού στα οποία δεν χρειάζεται να εξετάζει κανείς όλες τις στήλες (μεταβλητές) σε κάθε βήμα. Ήδη μέχρι σήμερα υπάρχει ένας μεγάλος αριθμός δημοσιευμένων εργασιών βασισμένες στην στρατηγική της επίλυσης ενός μοντέλου με μεγάλο αριθμό μεταβλητών χωρίς όμως να λαμβάνονται υπόψη κάθε φορά όλες οι μεταβλητές του προβλήματος.

Στη δυσκολία επίλυσης αυτών των μοντέλων έρχεται να δώσει λύση η τεχνική της **δυναμικής δημιουργίας μεταβλητών (column generation - CG)**. Η τεχνική αυτή επιτρέπει την επίλυση προβλημάτων με μεγάλο αριθμό μεταβλητών, χωρίς όμως να χρειάζεται να κατασκευαστούν εκ των προτέρων όλες οι μεταβλητές του προβλήματος. Πρόκειται για μία επαναληπτική διαδικασία κατά την οποία σε κάθε βήμα κατασκευάζεται ένας μικρός αριθμός μεταβλητών και με συστηματικά βήματα η διαδικασία οδηγεί τελικά στη βέλτιστη ή σε μία πολύ καλή λύση του συνολικού προβλήματος. Σήμερα μάλιστα, περισσότερο από κάθε άλλη φορά, αποτελεί την πρωτεύουσα προσέγγιση για επίλυση προβλημάτων με τεράστιο αριθμό μεταβλητών.

Αν και η βασική ιδέα της τεχνικής έμεινε η ίδια, το περιβάλλον υποστήριξης άλλαξε στο πέρασμα του χρόνου. Σήμερα το υλικό των υπολογιστών έχει βελτιωθεί φτάνοντας ή ακόμα και ξεπερνώντας την ισχύ των mainframes της προηγούμενης γενιάς. Η διαθέσιμη φυσική μνήμη των προσωπικών υπολογιστών αυξήθηκε

κατακόρυφα ενώ παράλληλα δημιουργήθηκαν πανίσχυροι επίλυτές γραμμικών αλλά και ακέραιων προβλημάτων. Έτσι, έγινε δυνατή η ενσωμάτωση της τεχνικής της δυναμικής δημιουργίας μεταβλητών σε σχήματα branch and bound και με αυτό τον τρόπο έγινε εφικτή η επίλυση μεγάλων προβλημάτων ακέραιου προγραμματισμού.

Τη χρήση του ακέραιου γραμμικού προγραμματισμού την προτείνουν αρκετοί συγγραφείς στη βιβλιογραφία για την επίλυση προβλημάτων χρονικού προγραμματισμού. Χαρακτηριστικά αναφέρονται οι παρακάτω: Breslaw (1976), Shin and Sullivan (1977), McClure and Wells (1984), Tripathy (1984), Ferland and Roy (1985), Tripathy (1992) [8-13]

3.3 Εφαρμογή τεχνικών Ροής Δικτύων

Η εφαρμογή της **Ροής Δικτύων (Network Flow)** έχει ως πυρήνα της τη χρήση ενός αλγορίθμου για το σχεδιασμό του ωρολογίου προγράμματος. Την τεχνική αυτή, πρότειναν μεταξύ άλλων οι Dyer and Mulvey (1976), Mulvey (1982), Chahal and de Werra (1989), και Dinkel (1989) [].

Το γενικό μοντέλο της τεχνικής ροής δικτύων διαμορφώνεται ως εξής:

$$\min \sum_{j=1}^m c_j x_j$$

με τους περιορισμούς:

$$\begin{aligned} Ax &= b \\ l &\leq x \leq u \end{aligned}$$

όπου $A_{n \times m}$ είναι ο κόμβος, $b_{n \times 1}$ είναι το διάνυσμα των προμηθειών, και $u_{1 \times m}$, $l_{1 \times m}$ είναι τα διανύσματα της δυναμικότητας και του κάτω ορίου αντίστοιχα.

Το δίκτυο που εφαρμόστηκε από το Denkel περιέχει τρία επίπεδα, μία πηγή και τον sink κόμβο. Το πρώτο επίπεδο αναφέρεται ως το επίπεδο των Τμημάτων, και περιέχει έναν κόμβο για κάθε ένα τμήμα, έτσι ώστε όλοι οι κόμβοι να συνδέονται με την πηγή. Το δεύτερο επίπεδο είναι το επίπεδο Καθηγητών/Προσωπικού που περιέχει έναν κόμβο για κάθε πιθανό συνδυασμό ενός καθηγητού και των μαθημάτων που διδάσκει αυτός ο καθηγητής. Αυτοί οι κόμβοι συνδέονται με τους κόμβους αντιπροσωπεύουν τα τμήματα στα οποία ανήκουν οι καθηγητές. Το τρίτο επίπεδο ονομάζεται επίπεδο Αιθουσών Χώρου/Χρόνου. Περιέχει έναν κόμβο για κάθε ένα

συνδυασμό αίθουσας με χρονική περίοδο. Κάθε ένας από τους κόμβους αυτού του επιπέδου συνδέεται με έναν κόμβο του δευτέρου επιπέδου μόνο στη περίπτωση που το μέγεθος της αίθουσας που εμφανίζεται στον κόμβο είναι συμβατό με τον αριθμό των φοιτητών του μαθήματος που εμφανίζεται στον άλλο κόμβο. Με μία γραμμή μεταξύ των επιπέδων δύο και τρία απεικονίζεται μία πιθανή διάλεξη.

Οι δυνατότητες και τα κάτω όρια των πλευρών του κόμβου που αντιπροσωπεύουν τις διαλέξεις, είναι 0 και 1 αντίστοιχα, και χάρη στην ύπαρξη της μονοτονίας, εξασφαλίζεται ότι η βέλτιστη λύση (optimal solution) θα έχει ακέραιες τιμές.

Οι συντελεστές της αντικειμενικής συνάρτησης προσδιορίζονται με βάση τη διαθεσιμότητα των καθηγητών και αιθουσών, καθώς επίσης και σύμφωνα με τις προτιμήσεις των καθηγητών. Το μοντέλο ροής δικτύου μπορεί να επιλυθεί σε πολυωνυμικό χρόνο, ωστόσο δεν εξασφαλίζει ότι η λύση που θα προκύψει θα είναι εφικτή. Για παράδειγμα, δεν θα τοποθετεί στο ωρολόγιο πρόγραμμα έναν καθηγητή σε πολλαπλές διαλέξεις την ίδια ώρα. Για το λόγο αυτό εάν η διαδικασία οδηγηθεί σε μία εφικτή λύση, ολοκληρώνεται. Διαφορετικά ο χρήστης πρέπει να παρέμβει και «με το χέρι» να αλλάξει κάποιες από τις σταθμίσεις ώστε να ξεπεραστεί η αιτία που παράγει μη εφικτές λύσεις στο πρόβλημα. Η διαδικασία εκτελείται αρκετές φορές, ώσπου εξασφαλιστεί μία εφικτή λύση.

3.4 Αναγωγή σε πρόβλημα Χρωματισμού Γραφήματος

Το **πρόβλημα χρωματισμού γραφήματος (graph coloring)** είναι ένα από τα κλασικά NP-complete προβλήματα γραφημάτων (Garey and Johnson, 1979) [6]. Η χρήση αυτής της τεχνικής για την επίλυση προβλημάτων χρονικού προγραμματισμού προτείνεται από τον De Werra (1985) [5] και περιγράφεται ως εξής [1]:

Σχεδιάζουμε ένα κόμβο (vertex) m_{ij} για κάθε διάλεξη l_i κάθε μαθήματος K_j . Για κάθε μάθημα K_j θέτουμε μία κλίκα-ομάδα (clique) ανάμεσα στους κόμβους m_{ij} (για $i = 1, \dots, q$) που εκφράζουν τις διαλέξεις του μαθήματος. Ενώνουμε με μία γραμμή την ομάδα του μαθήματος K_{j1} και την ομάδα του μαθήματος K_{j2} , εάν πρόκειται για μαθήματα K_{j1} και K_{j2} που επικαλύπτονται- συγκρούονται.

Στην περίπτωση που υπάρχει κάποια έλλειψη διαθεσιμότητας, εισάγουμε μία σειρά από p νέους κόμβους, έτσι ώστε κάθε ένας να αντιστοιχεί σε μία χρονική περίοδο. Οι νέοι κόμβοι συνδέονται όλοι μεταξύ τους. Αυτό εξασφαλίζει ότι ο κάθε νέος κόμβος προσδιορίζεται με ένα διαφορετικό χρώμα.

Εάν ένα μάθημα δεν μπορεί να έχει τις διαλέξεις του σε μία δεδομένη περίοδο, τότε όλοι οι κόμβοι που αντιστοιχούν στις διαλέξεις του συνδέονται με τον κόμβο που αντιστοιχεί στην συγκεκριμένη περίοδο. Αντιθέτως, εάν μία διάλεξη πρέπει να λάβει χώρα μία συγκεκριμένη περίοδο, τότε οι κόμβοι που αντιστοιχούν σε αυτή την ομάδα συνδέονται με τους κόμβους σε όλες τις χρονικές περιόδους, εκτός από τη συγκεκριμένη περίοδο που επιθυμούμε να τοποθετήσουμε τη διάλεξη. Ο ελάχιστος αριθμός διαφορετικών χρωμάτων που απαιτούνται για τον χρωματισμό του γραφήματος καλείται χρωματικός αριθμός. Αν ο χρωματικός αριθμός είναι ίσος ή μικρότερος από τον αριθμός των διαθέσιμων χρονικών περιόδων, σημαίνει ότι θα μπορέσουμε να σχεδιάσουμε ένα ωρολόγιο πρόγραμμα χωρίς συγκρούσεις-επικαλύψεις μαθημάτων.

Την αναγωγή του προβλήματος ωρολόγιου προγραμματισμού σε πρόβλημα χρωματισμού γραφήματος εφαρμόζει, μεταξύ άλλων και ο Selim (1988) [7].

Στο Παράρτημα υπάρχει η επίλυση ενός απλού παραδείγματος προγραμματισμού διαλέξεων με τη μέθοδο του χρωματισμού γραφήματος για τη διαγραμματική απεικόνιση της μεθόδου (Παράρτημα 2).

3.5 Local Search based techniques

Οι **local search techniques** ανήκουν στις πιο επιτυχημένες προσεγγίσεις επίλυσης προβλημάτων βελτιστοποίηση τα τελευταία χρόνια και συγκαταλέγονται στις μετα-ευρετικές μεθόδους (meta-heuristics). **Ο όρος Local Search αποτελεί στην ουσία την ονομασία ενός συνόλου μεθόδων οι οποίες επαναλαμβάνουν την αντικατάσταση μίας τρέχουσας λύσης s από μία νέα λύση s^* , έως οι συνθήκες για διακοπή της διαδικασίας που έχουμε θέσει ικανοποιηθούν.** Η νέα λύση επιλέγεται από ένα σύνολο $N(s)$. Το σύνολο $N(s)$ αποτελεί το σύνολο των υποψήφιων λύσεων

και μέσα σε αυτό το σύνολο η τρέχουσα λύση μπορεί να μετασχηματιστεί, συνήθως με μία μόνο κίνηση.

Η **ποιότητα** της λύση χαρακτηρίζεται από την απόδοσή της, δηλαδή την τιμή που δίνει στην **αντικειμενική συνάρτηση κόστους $f(s)$** . Στόχος της ερευνητικής διαδικασίας είναι η ελαχιστοποίηση της αντικειμενικής συνάρτησης.

Υπάρχουν διαφορετικές παραλλαγές της βασικής μεθόδου Local Search. Η διαφορά στις μεθόδους αυτές έγκειται στους διαφορετικούς μηχανισμούς αποδοχής ή απόρριψης μίας υπονήφιας λύσης από το σύνολο $N(s)$, τον ορισμό του συνόλου $N(s)$ και των συνθηκών διακοπής της διαδικασίας. [2]

Στη συνέχεια του κεφαλαίου παρατίθενται κάποιες από τις μεθόδους επίλυσης προβλημάτων ωρολόγιου προγραμματισμού, που ανήκουν στην κατηγορία των Local Search Techniques.

3.5.1 Hill Climbing

Μία από τις πιο απλές σε δομή και εφαρμογή μεθόδους επίλυσης προβλημάτων βελτιστοποίησης είναι η τεχνική **Hill Climbing**. Η Hill Climbing ανήκει στις Local Search Techniques και χρησιμοποιήθηκε για την επίλυση προβλημάτων ωρολόγιου προγραμματισμού ήδη από το 1960 από τους Appleby et al [4].

Όπως προαναφέραμε η τεχνική αυτή βασίζεται σε έναν πολύ απλό και ευκολονόητο αλγόριθμο. Μία λύση γίνεται αποδεκτή ως νέα λύση s^* εάν παρέχει καλύτερη ή ίση απόδοση σε σχέση με την τρέχουσα λύση s . Η τεχνική Hill Climbing δεν απαιτεί τον ορισμό καμίας παραμέτρου και η συμπεριφορά του αλγορίθμου είναι ιδιαίτερος σταθερή. Σκοπός του αλγορίθμου είναι να συγκλίνει σε μία λύση πολύ σύντομα, κάτι όμως που πολύ συχνά οδηγεί σε μία όχι ιδιαίτερα ποιοτική τελική λύση. Η χαμηλής ποιότητας λύση ερμηνεύεται ως σημαντική απόκλιση-απόσταση της λύσης μας από τη βέλτιστη λύση του προβλήματος.

3.5.2. Simulated Annealing

Μία ακόμα μέθοδος που ανήκει στις Local Search Techniques είναι η **Simulated Annealing**. Η μέθοδος αυτή είναι μετα-ευρετική και χρησιμοποιεί τις πιθανότητες. Την πρότεινε ο Kirkpatrick το 1983 για την επίλυση προβλημάτων

βελτιστοποίησης. Η ονομασία της προέρχεται από το γεγονός ότι η μέθοδος αρχικά χρησιμοποιήθηκε για να προσομοιώσει τη διαδικασία ψύξης μίας ομάδας θερμών δονούμενων ατόμων. Η μέθοδος αυτή κατά βάση είναι πολύ σχετική και εμφανίζει σημαντικές ομοιότητες με την Hill Climbing. Επειδή όμως εξασφαλίζει καλύτερες ποιοτικά λύσεις στο χρήστη, λόγω του ότι διευρύνει το ερευνητικό πεδίο, είναι αρκετά πιο διαδεδομένη στην επίλυση προβλημάτων βελτιστοποίησης.

Κεντρική ιδέα αυτής της μεθόδου είναι η αποδοχή χειρότερων λύσεων με πιθανότητα $P = e^{-\delta/T}$, όπου $\delta = f(s^*) - f(s)$ και ως T ορίζεται η Θερμοκρασία.

Η τεχνική της Simulated Annealing ξεκινά με την δημιουργία μιας τυχαίας αρχικής λύσης s_{init} . Η κύρια διαδικασία αποτελείται από μία ακολουθία επαναλαμβανόμενων εντολών που δημιουργεί τυχαία σε κάθε επανάληψη, ένα σύνολο τιμών $N(s)$ της τρέχουσας λύσης. Ο ορισμός αυτού του συνόλου τιμών εξαρτάται από την δομή του εκάστοτε προβλήματος.

Ας θέσουμε Δ την διαφορά στην τιμή της αντικειμενικής συνάρτησης μεταξύ της νέας λύσης s^* και της τρέχουσας λύσης s σε ένα πρόβλημα ελαχιστοποίησης. **Εάν $\Delta < 0$, η νέα λύση s^* γίνεται αποδεκτή και τίθεται η νέα τρέχουσα λύση. Εάν $\Delta \geq 0$, η νέα λύση s^* γίνεται αποδεκτή με πιθανότητα $e^{-\delta/T}$.** Η θερμοκρασία αρχικά τίθεται σε ένα ανώτατο όριο T_0 . Μετά από έναν ορισμένο αριθμό επαναλήψεων η θερμοκρασία μειώνεται από το ποσοστό ‘ψύξης’ α , έτσι ώστε $T_n = \alpha \times T_{n-1}$, όπου $0 \leq \alpha \leq 1$.

Η διαδικασία διακόπτεται όταν η θερμοκρασία φτάσει σε τιμή πολύ κοντά στο 0 και καμία λύση που αυξάνει την τιμή της αντικειμενικής συνάρτησης δεν γίνεται πια αποδεκτή, δηλαδή το σύστημα «παγώνει». Η λύση που αποκτάται όταν το σύστημα έχει παγώσει, είναι προφανώς ένα τοπικό ελάχιστο (Local minimum).

Τα στοιχεία ελέγχου της διαδικασίας είναι το ποσοστό ‘ψύξης’ α , ο αριθμός των επαναλήψεων σε κάθε θερμοκρασία και η αρχική θερμοκρασία T_0 .

3.5.3. The Threshold Acceptance Algorithm

Η τεχνική του **Threshold Acceptance αλγόριθμου**, αποτελεί μία παραλλαγή της μεθόδου Simulated Annealing και παρουσιάστηκε για πρώτη φορά από τους Dueck και Scheuer του 1989 []. Ο αλγόριθμος που χρησιμοποιεί η μέθοδος είναι ο

αλγόριθμος που χρησιμοποιείται και για την επίλυση του προβλήματος του πλανόδιου πωλητή (Travelling Salesman Problem).

Η επίλυση προβλημάτων βελτιστοποίησης με τη χρήση αυτή της μεθόδου γίνεται βασιζόμενη στη **αποδοχή κάθε χειρότερης λύσης**, όταν η διαφορά $\delta = f(s^*) - f(s)$ **δεν υπερβαίνει το όριο αποδοχής T** (ορισμένο από το χρήστη).

Αρχικά οι εμπνευστές της μεθόδου πρότειναν να μειώνεται το όριο αποδοχής T όταν ο αλγόριθμος δεν δίνει καλύτερη λύση για μεγάλο χρονικό διάστημα. Ωστόσο κάτι τέτοιο δεν είναι απλό και δυσκολεύει την πρακτική εφαρμογή της μεθόδου, καθώς δεν είναι ξεκάθαρο πότε να μειωθεί το όριο αποδοχής T και πόση να είναι η μείωση που θα υποστεί. Παρ' όλη όμως τη δυσκολία που προέκυψε αρχικά κατά την εφαρμογή το αλγορίθμου, η διαδικασία αποδοχής βελτιώθηκε αλλά ακόμα περιλαμβάνει κάποιες παραμέτρους των οποίων η τιμή πρέπει να καθοριστεί εμπειρικά.

3.5.4. Great Deluge Algorithm

Η τεχνική επίλυσης προβλημάτων βελτιστοποίησης που στηρίζεται στη χρήση του **Great Deluge αλγορίθμου** και εισήχθη για πρώτη φορά από τον Dueck το 1993. Ο συγκεκριμένος αλγόριθμος αποδέχεται κάθε λύση η οποία δίνει τιμή στην αντικειμενική συνάρτηση μικρότερη ή ίση σε σχέση με το ανώτατο όριο B που έχουμε θέσει. Η τιμή του B εμφανίζει φθίνουσα μονοτονία κατά τη διάρκεια της έρευνας και οριοθετεί την εφικτή περιοχή του ερευνητικού χώρου, την περιοχή δηλαδή που περιλαμβάνει το σύνολο όλων των πιθανών λύσεων του προβλήματος βελτιστοποίησης. Συνήθως, ο αλγόριθμος συγκλίνει όταν προκύπτει μία τρέχουσα λύση η οποία ξεπερνά το ανώτατο όριο. Προκειμένου να προληφθεί μία τέτοια σύγκλιση (ώστε να επαναφέρει την τρέχουσα λύση στην εφικτή περιοχή) και να βελτιωθεί η απόδοση της μεθόδου, προτείνεται η επέκταση του αλγορίθμου ώστε να αποδέχεται κάθε υποψήφια λύση που είναι καλύτερη από την τρέχουσα λύση s. Ο ψευδοκώδικας που επιφέρει την επέκταση του αλγορίθμου παρατίθεται παρακάτω:

Θέστε την αρχική λύση s

Υπολογίστε την τιμή της αρχικής αντικειμενικής συνάρτησης $f(s)$

Αρχικό όριο $B_0 = f(s)$

Καθορίστε τις παραμέτρους εισόδου $\Delta B = ?$

Ενώ δεν ικανοποιούνται κάποιες συνθήκες διακοπής **κάνε**

 Προσδιορίστε το σύνολο τιμών $N(s)$

 Τυχαία επιλέξτε μία υποψήφια λύση $s^* \in N(s)$

 Εάν $(f(s^*) \leq f(s))$ or $(f(s) \leq B)$

 Τότε **αποδοχή s^***

 Χαμήλωσε το ανώτατο όριο $B = B_0 - \Delta B$

Σχήμα 1: Επέκταση του Great Deluge αλγορίθμου

3.5.5 Tabu Search

Η **Tabu Search** μετα-ευρετική μέθοδος επίλυσης προβλημάτων χρονικού προγραμματισμού που δίνει καλά ποιοτικών αποτελέσματα και ανήκει στις Local Search Techniques. Η μέθοδος αυτή χρησιμοποιεί έναν αλγόριθμο, ο οποίος εξερευνά ένα υποσύνολο V του συνόλου $N(s)$ της τρέχουσας λύσης s . Ο ορισμός του συνόλου τιμών $N(s)$, όπως και στη μέθοδο Simulated Annealing, εξαρτάται από την δομή του εκάστοτε προβλήματος. **Η λύση του υποσυνόλου V , η οποία δίνει την χαμηλότερη τιμή στην αντικειμενική συνάρτηση κόστους, γίνεται η νέα τρέχουσα λύση s^* , ανεξάρτητα από εάν η τιμή που προτείνεται να γίνει τρέχουσα είναι καλύτερη ή χειρότερη της τιμής s .**

Για να αποφευχθεί η «ανακύκλωση» των λύσεων, ορίζεται μία λίστα που περιλαμβάνει τις λύσεις τις οποίες ο αλγόριθμος έχει ήδη υπολογίσει και στις οποίες απαγορεύεται να επιστρέψει. Είναι η αποκαλούμενη tabu-list. Περιλαμβάνει τις τελευταίες k τρέχουσες λύσεις, όπου k είναι μία παράμετρος της μεθόδου, και λειτουργεί σαν «ουρά» έτσι ώστε όποτε προστίθεται μία νέα λύση, λόγω της κίνησης και της εισόδου της νέας λύσης στη λίστα, η παλαιότερη λύση αποβάλλεται.

Εάν μία λύση προκαλεί μεγάλη βελτίωση στην τιμή της αντικειμενικής συνάρτησης, η tabu status παύει και η λύση γίνεται αποδεκτή ως η νέα σωστή λύση. Πιο συγκεκριμένα, ορίζουμε μία **aspiration** συνάρτηση A η οποία για κάθε τιμή της

αντικειμενικής συνάρτησης, μας δίνει τιμή για τη συνάρτηση A , η οποία αντιπροσωπεύει το πόσο απέχει η τιμή που προέκυψε από τον αλγόριθμο (δηλαδή την επιδιωκόμενη τιμή). Δεδομένης μίας τρέχουσας λύσης s , με αντικειμενική συνάρτηση την f , και με καλύτερη λύση συνόλου την s' , εάν $f(s') < A(f(s))$ τότε η λύση s' τίθεται η νέα τρέχουσα λύση s^* , ακόμα και εάν η λύση s' αποτελεί κίνηση του αλγορίθμου (tabu move).

Η διαδικασία σταματά σε δύο περιπτώσεις. Πρώτον, όταν γίνουν όλες οι επαναλήψεις που επιτρέπονται σύμφωνα με την τιμή έχει δοθεί και δεύτερον όταν η τιμή της αντικειμενικής συνάρτησης στην τρέχουσα λύση γίνει ίση με το καθορισμένο κατώτατο όριο.

Οι παράμετροι ελέγχου της διαδικασίας καθώς και το μέγεθος της tabu λίστας k , η aspiration συνάρτηση A , και τα στοιχεία του συνόλου V της neighbor των λύσεων ελέγχεται σε κάθε μία επανάληψη που πραγματοποιείται.

Πιο πρόσφατες μελέτες πάνω στο σχεδιασμό του ωρολόγιου προγράμματος μαθήματος πανεπιστημίων, συνδυάζουν την Tabu Search με άλλες μεθόδους αναζήτησης. Ο Schaerf εφαρμόζει ένα συνδυασμό της Hill-Climbing με την Tabu Search, ενώ οι Burke et al. χρησιμοποιούν ένα υπερ-ευρετικό (Hyperheuristic) στο οποίο ο αλγόριθμος της Tabu Search εργάζεται στην επιλογή χαμηλού επιπέδου ευρετικών.

3.6 Εξελικτικοί αλγόριθμοι (Evolutionary algorithms)

Οι εξελικτικοί αλγόριθμοι αποτελούν τεχνικές επίλυσης προβλημάτων χρονικού προγραμματισμού βάσει πληθυσμών (Population based techniques). Οι τεχνικές αυτές χρησιμοποιούν τους Γενετικούς, του Μιμητικούς και τους Ant αλγορίθμους.

3.6.1. Γενετικοί αλγόριθμοι (Genetic algorithms)

Έχει ήδη αναφερθεί ότι οι **Γενετικοί αλγόριθμοι** ανήκουν στις μετά-ευρετικές (meta-heuristics) μεθόδους επίλυσης προβλημάτων χρονικού προγραμματισμού. Οι Γενετικοί αλγόριθμοι ανήκουν στο κλάδο της επιστήμης των υπολογιστών και αποτελούν μια ακόμα μέθοδο αναζήτησης βέλτιστων λύσεων σε

προβλήματα χρονικού προγραμματισμού. Το σημαντικό πλεονέκτημα των Γενετικών αλγορίθμων είναι ότι μπορούν να επεξεργαστούν και να δώσουν λύσεις σε προβλήματα που περιέχουν πολλές παραμέτρους. Η ύπαρξη πολλών παραμέτρων αποτελεί σοβαρό εμπόδιο για τις άλλες αναλυτικές μεθόδους με αποτέλεσμα να μη μπορούν να βρουν το βέλτιστο συνδυασμό τιμών για τις μεταβλητές ώστε το υπό εξέταση σύστημα να αντιδρά με όσο το δυνατόν με το θεμιτό τρόπο σε ένα πρόβλημα με πολλές παραμέτρους.

Όπως φαίνεται και από την ονομασία των Γενετικών αλγορίθμων, ο τρόπος λειτουργίας τους και η δομή τους είναι εμπνευσμένοι από την επιστήμη της βιολογίας. Πιο συγκεκριμένα, οι αλγόριθμοι έχουν ως κεντρική ιδέα τη θεωρία της εξέλιξης των ειδών, η οποία πραγματοποιείται μέσω της γενετικής μετάλλαξης, της φυσικής επιλογής και της διασταύρωσης.

Ένα ακόμα ουσιαστικό πλεονέκτημα των Γενετικών αλγορίθμων είναι η αρκετά απλή υλοποίησή τους. Οι παράμετροι του προβλήματος κωδικοποιούνται με τέτοιο τρόπο έτσι ώστε να αναπαρασταθούν από μία μεταβλητή που αποτελείται από σειρά χαρακτήρων ή δυαδικών ψηφίων. Η μεταβλητή αυτή μιμείται τις διαδικασίες του γενετικού κώδικα που υπάρχει στους ζωντανούς οργανισμούς.

Στην εκκίνηση της διαδικασίας επίλυσης ο Γενετικός αλγόριθμος παράγει πολλαπλά αντίγραφα της μεταβλητής, συνήθως με τυχαίες τιμές, δημιουργώντας ένα πληθυσμό-σύνολο λύσεων. Κάθε λύση, η οποία εκφράζει τιμές για τις παραμέτρους του προβλήματος, δοκιμάζεται ώστε να εξεταστεί ο βαθμός στο οποίο δίνει στο πρόβλημα την επιθυμητή αντίδραση. Η απόσταση από την επιθυμητή αντίδραση του προβλήματος υπολογίζεται μέσω της συνάρτησης ικανότητας, που μας δίνει το μέτρο ικανότητας της λύσης.

Οι λύσεις οι οποίες βρίσκονται περισσότερο κοντά στην επιθυμητή, σε σχέση με τις υπόλοιπες λύσεις, και σύμφωνα πάντα με το μέτρο που υπολογίζεται από τη συνάρτηση ικανότητας, αναπαράγονται στην επόμενη γενιά λύσεων και λαμβάνουν μία τυχαία «μετάλλαξη». Η διαδικασία επαναλαμβάνεται για αρκετές φορές («γενιές»). Οι τυχαίες «μεταλλάξεις» σε συνδυασμό με την «επιβίωση» και την αναπαραγωγή των λύσεων («γονιδίων») που πλησιάζουν καλύτερα το επιθυμητό αποτέλεσμα, θα παράγουν τη λύση που θα περιέχει τις τιμές για τις παραμέτρους που θα ικανοποιούν όσο το δυνατόν καλύτερα τη συνάρτηση ικανοποίησης. Για να οδηγηθούμε πιο γρήγορα στο αποτέλεσμα υπάρχουν κάποιες ακόμα εκδοχές της

προαναφερθείσας διαδικασίας στις οποίες περιλαμβάνονται διασταυρώσεις των λύσεων («γονιδίων»).

Στην πραγματικότητα οι Γενετικοί αλγόριθμοι προσπαθούν να βρουν τη λύση ενός προβλήματος με το να προσομοιώνουν την εξέλιξη ενός «πληθυσμού λύσεων» του προβλήματος.

Ένα ακόμα χαρακτηριστικό των Γενετικών αλγορίθμων είναι ότι λόγω της ύπαρξης του παράγοντα της τύχης (τυχαία μετάλλαξη και τυχαίο ζευγάρωμα) κάθε εκτέλεση του αλγορίθμου είναι πιθανό να συγκλίνει σε διαφορετική λύση και σε διαφορετικό χρόνο.

Η απόδοση των Γενετικών αλγορίθμων εξαρτάται ως επί το πλείστον από την συνάρτηση ικανότητας και συγκεκριμένα από το κατά πόσο το μέτρο της περιγράφει την βέλτιστη λύση.

3.6.2 Μιμητικοί αλγόριθμοι (Memetic algorithms)

Οι **Μιμητικοί αλγόριθμοι** αποτελούν μία προσέγγιση επίλυσης που βασίζεται στον πληθυσμό και ανήκουν στις μετά-ευρετικές (meta-heuristics) μεθόδους επίλυσης προβλημάτων χρονικού προγραμματισμού και γενικότερα βελτιστοποίησης. Οι Μιμητικοί αλγόριθμοι είναι τάξεις μεγέθους πιο ταχείς από τους Γενετικούς αλγορίθμους σε κάποια προβλήματα.

Ουσιαστικά συνδυάζουν τις ευρετικές Local Search μεθόδους με τη διαδικασία της διασταύρωσης. Για το λόγο αυτό, ορισμένοι ερευνητές αποκαλούν τους Μιμητικούς αλγορίθμους, Ιβρυδικούς Γενετικούς αλγορίθμους (Hybrid Genetic Algorithm). Η μέθοδος των μιμητικών αλγορίθμων γνωρίζει αυξανόμενη αποδοχή σε πολλά προβλήματα βελτιστοποίησης και σε κάποιες περιπτώσεις δίνει βέλτιστες λύσεις ενώ οι άλλες μετα-ευρετικές (metaheuristics) αποτυγχάνουν.

3.6.3. Ant algorithms

Οι **Ant αλγόριθμοι** προσομοιώνουν τον τρόπο που τα μυρμήγκια αναζητούν την πιο σύντομη διαδρομή προς την τροφή τους, εναποθέτοντας φερεμόνη στο δρόμο τους. Οι πιο σύντομες διαδρομές παράγουν ισχυρότερα επίπεδα φερεμόνης σε μία συγκεκριμένη χρονική περίοδο.

Στον αλγόριθμο, κάθε μυρμήγκι χρησιμοποιείται για να κατασκευαστεί μία λύση και η πληροφορία κατά τη διάρκεια της αναζήτησης διατηρείται ως φερεμόνη, η οποία χρησιμοποιείται για να βοηθήσει στην παραγωγή λύσεων στο επόμενο στάδιο.

3.7 Τεχνικές επίλυσης βάσει περιορισμών (Constraint based techniques)

Οι τεχνικές που επιλύουν προβλημάτων σχεδιασμού ωρολόγιου προγράμματος βάσει περιορισμών προέρχονται από το χώρο της Τεχνητής Νοημοσύνης και είναι οι ακόλουθες: **λογικός προγραμματισμός περιορισμών (constraint logic programming)** και **τεχνικές ικανοποίησης περιορισμών (constraint satisfaction techniques)**. Αυτές μέθοδοι γοήτευσαν τους ερευνητές των ωρολόγιων προγραμμάτων λόγω της ευκολίας και της ευελιξίας στη εφαρμογή και λειτουργία τους σε αυτού του είδους τα προβλήματα.

Οι τεχνικές επίλυσης βάσει περιορισμών είναι συνήθως υπολογιστικά ακριβές επιλογές λόγω του αριθμού των πιθανών εκθετικών αυξήσεων των εντολών, σε σχέση με τον αριθμό των μεταβλητών.

3.7.1. Constraint logic programming

Το πρόβλημα του ωρολόγιου προγραμματισμού μπορεί να μορφοποιηθεί ως ένα πρόβλημα ικανοποίησης περιορισμών (Constraint Satisfaction problem CSP). Σε τέτοιο πρόβλημα πρέπει να βρεθούν οι τιμές που να ικανοποιούν ένα σύνολο περιορισμών για ένα σύνολο διακεκριμένων μεταβλητών, με πεπερασμένο πεδίο ορισμού.

Ο λογικός προγραμματισμός περιορισμών (Constraint Logic programming CLP) βασίζεται στην εφαρμογή επεξηγηματικών γλωσσών λογικού προγραμματισμού (declarative logic programming languages) όπως Prolog to CSPs. Ένα Prolog πρόγραμμα αποτελείται από ένα σύνολο προτάσεων, όπου στην περίπτωση των CLP, περιέχουν περιορισμούς, η ικανοποίηση των οποίων ελέγχεται κατά τη διάρκεια της διαδικασίας υπολογισμού.

Ειδικότερα η εφαρμογή του CLP συστήματος σε προβλήματα βελτιστοποίησης, όπως αυτό που πραγματευόμαστε, προσφέρει μία τεχνική επίλυσης βασισμένη σε μία εκτενή πρωταρχική branch and bound αναζήτηση.

3.8 Πολυκριτηριακή ανάλυση (Multi-Criteria techniques)

Με διατύπωση του προβλήματος του σχεδιασμού του ωρολόγιου προγράμματος μαθημάτων ενός πανεπιστημίου ως πρόβλημα πολυκριτηριακής ανάλυσης θέτουμε κριτήρια τα οποία μετρούν την παραβίαση των αντίστοιχων περιορισμών.

Μία γενική μορφοποίηση ενός προβλήματος βελτιστοποίησης σύμφωνα με την πολυκριτηριακή ανάλυση είναι:

$$\text{Ελαχιστοποίηση ή Μεγιστοποίηση } F(x) = (f_1(x), f_2(x), \dots, f_k(x))$$

$$\text{μ.τ.π. } x \in S$$

Όπου το x είναι μία λύση, S το σύνολο των εφικτών λύσεων, k ο αριθμός των αντικειμενικών στόχων στο πρόβλημα, $F(x)$ η απεικόνιση της λύσης x στο k -πολυνυμικό χώρο και κάθε $f_i(x)$, το $i = 1, \dots, k$ αντιπροσωπεύει ένα αντικειμενικό στόχο (ελαχιστοποίησης ή μεγιστοποίησης).

Για την επίλυση του προβλήματος του ωρολόγιου προγραμματισμού η διατύπωση, δεδομένου του αριθμού των χρονικών περιόδων P , του αριθμού των μαθημάτων N και του αριθμού των απαιτούμενων διαλέξεων-εργαστηρίων, προσδιορίζει ένα ωρολόγιο πρόγραμμα T (Timetable) δίνοντας τη μικρότερη δυνατή τιμή σε όλα τα μέρη του παρακάτω διανύσματος:

$$WF = (w_1 f_1(T), \dots, w_k f_k(T), \dots, w_K f_K(T))$$

σύμφωνα του περιορισμούς:

$$\sum_{p=1}^P t_{np} = l_n, n = 1, \dots, N \quad (1)$$

$$\sum_{n=1}^N t_{np} \leq r_p, p = 1, \dots, P \quad (2)$$

$$\sum_{n=1}^{N-1} \sum_{m=n+1}^N \sum_{p=1}^P t_{np} t_{mp} c_{nm} = 0 \quad (3)$$

Όπου,

- ✓ **N** ο αριθμός των μαθημάτων.
- ✓ **P** ο αριθμός των διαθέσιμων χρονικών περιόδων.
- ✓ **I_n** ο αριθμός των διαλέξεων των μαθημάτων n, n = 1, ..., N, που πρέπει να τοποθετηθούν κατά τη διάρκεια των P χρονικών περιόδων.
- ✓ **r_p** ο αριθμός των διαθέσιμων αιθουσών την περίοδο p, p = 1, ..., P.
- ✓ **C = [c_{nm}]_{NxN}** ο συμμετρικός πίνακας που αναπαριστά τα συγκρουόμενα-επικαλυπτόμενα μαθήματα.
- ✓ **c_{nm}** ο αριθμός των φοιτητών που παρακολουθούν και τα δύο μαθήματα n και m, όπου n, m = 1, ..., N.
- ✓ **K** ο αριθμός των κριτηρίων.
- ✓ **T = [t_{np}]_{NxP}** ο πίνακας που περιγράφει τις τοποθετήσεις των μαθημάτων στις χρονικές περιόδους.
- ✓ **t_{np}** $\begin{cases} 1, & \text{εάν το μάθημα n προγραμματίστηκε στην χρονική περίοδο p.} \\ 0, & \text{σε διαφορετική περίπτωση.} \end{cases}$

όπου n = 1, ..., N, p = 1, ..., P.

- ✓ **f_k(T)** η τιμή των κριτηρίων C_k, k = 1, ..., K.
- ✓ **w_k** το βάρος, η στάθμιση των κριτηρίων C_k, k = 1, ..., K.

Οι περιορισμοί (1), (2), (3), εξασφαλίζουν ότι κάθε μάθημα έχει τις απαιτούμενες διαλέξεις-εργαστηρίων, ότι κάθε διάλεξη-εργαστήριο τοποθετείται σε συγκεκριμένες αίθουσες, και τα μαθήματα που έχουν κοινούς φοιτητές δεν προγραμματίζονται την χρονική περίοδο.

Συνήθως, ο σκοπός της επίλυσης είναι να επιτευχθεί η βελτιστοποίηση μίας ομάδας «οντοτήτων» του προβλήματος που οι επιπρόσθετες απαιτήσεις και οι περιορισμοί του προβλήματος να ικανοποιούνται.

Σε ένα πολυκριτηριακό πρόβλημα ποικίλλα κριτήρια υπάρχουν για να αξιολογούν την ποιότητα των λύσεων και σε κάθε ένα από αυτά τίθεται ένας αντικειμενικός στόχος (ελαχιστοποίησης ή μεγιστοποίησης). Συχνά παρατηρείται κάποια κριτήρια να είναι αντικρουόμενα κατά τρόπο που η βελτίωση ενός κριτηρίου να επιτυγχάνουν μόνο με την εκτεταμένη χειροτέρευση κάποιου άλλου. Επίσης μερικά από τα κριτήρια μπορεί να μην είναι συγκρίσιμα ποιοτικά στοιχεία που δυσκολεύει την επίλυση του προβλήματος.

Η πρώτη απόφαση που καλούμαστε να πάρουμε όταν χειριζόμαστε ένα πολυκριτηριακό πρόβλημα βελτιστοποίησης, είναι πώς θα συνδυάσουμε την αναζήτηση-έρευνα με τη διαδικασία λήψης απόφασης, την επιλογή δηλαδή της λύσης. Στη βιβλιογραφία προτείνονται τρεις τρόποι []:

- **Λήψη απόφασης και έπειτα αναζήτηση (Decision-making and then search) ‘a priori approach’:** Οι προτιμήσεις για κάθε αντικειμενικό στόχο τίθενται από τη διαδικασία λήψης απόφασης και έπειτα αναζητούνται οι λύσεις που να ικανοποιούν αυτές τις προτιμήσεις.
- **Αναζήτηση και ακολούθως λήψη απόφασης (Search and then decision-making) ‘a posteriori approach’:** Διάφορες λύσεις υπολογίζονται αρχικά και στη συνέχεια η διαδικασία λήψης απόφασης επιλέγει την πιο ικανοποιητική. Οι λύσεις που παρουσιάζονται, πρέπει να αντιπροσωπεύουν μία εναλλαγή μεταξύ των διάφορων αντικειμενικών στόχων.
- **Αλληλεπίδραση της αναζήτησης και της λήψης αποφάσεων (Interactive search and decision-making):** Η διαδικασία λήψης αποφάσεων αλληλεπιδρά κατά τη διάρκεια της αναζήτησης για να την οδηγεί σε υποσχόμενες λύσεις εκπληρώνοντας τις προτιμήσεις της διαδικασίας επίλυσης του προβλήματος.

4^ο Κεφάλαιο

Πολυκριτηριακή ανάλυση βάσει συναινετικού προγραμματισμού

4.1 Εισαγωγή

Στο κεφάλαιο που ακολουθεί, για την καλύτερη κατανόηση της διαδικασίας επίλυσης ενός προβλήματος χρονικού προγραμματισμού και ειδικότερα της κατάρτισης του ωρολόγιου προγράμματος μαθημάτων ενός εκπαιδευτικού ιδρύματος, κρίνεται σκόπιμο να μελετηθεί η εφαρμογή μίας εκ των μεθοδολογιών που παρουσιάστηκαν, σε πραγματικά δεδομένα. Είναι σημαντικό εκτός της θεωρητικής προσέγγισης, να παρουσιαστεί έστω και ακροθιγώς η πρακτική διάσταση του προβλήματος, έτσι όπως το αντιμετωπίζουν καθημερινά τα εκπαιδευτικά ιδρύματα σε όλο τον κόσμο. Καθώς, έχουμε ήδη αναφέρει ότι, το ζήτημα του χρονικού προγραμματισμού, συνδέεται άρρηκτα με όλα τα εκπαιδευτικά ιδρύματα και αποτελεί αναγκαία και σε μεγάλο βαθμό ικανή συνθήκη για την εύρυθμη και αποτελεσματική λειτουργία τους.

Η εξεύρεση των δεδομένων των προβλημάτων, που θα επιλυθούν στη συνέχεια του κεφαλαίου, σύμφωνα με την μέθοδο του συναινετικού προγραμματισμού, έγινε από τη βάση προβλημάτων OR-Library, J E Beasley που βρίσκεται στον ακόλουθο δικτυακό τόπο:

<http://people.brunel.ac.uk/~mastjjb/jeb/orlib/tableinfo.html>.

Πιο συγκεκριμένα, έχουν επιλεγεί δύο από τα πέντε προβλήματα, για τα οποία μας παρέχει τα δεδομένα και τις κατάλληλες πληροφορίες, η OR-Library, J E Beasley.

4.2 Επιλογή μεθοδολογίας

Όπως έχουμε ήδη αναφέρει, η πολυκριτηριακή προσέγγιση επίλυσης, αξιολογεί μία λύση σύμφωνα με τους περιορισμούς που έχουν τεθεί. Η αντικειμενική συνάρτηση του προβλήματος, εκτιμά και προσδιορίζει το βαθμό παραβίασης των περιορισμών. Το γεγονός αυτό απαιτεί τον καθορισμό κατάλληλων σταθμίσεων (συντελεστών βαρύτητας) για το πλήθος των περιορισμών που χρησιμοποιούν διαφορετικές μονάδες μετρήσεις, ώστε να είναι εφικτή η σύγκριση και ο υπολογισμός των παραβιάσεων τους. Επιπλέον, επειδή συνήθως η αντικειμενική συνάρτηση είναι προσθετική (additive), δεν είναι ιδιαιτέρως «εύκαμπτη», υπό την έννοια ότι επιτρέπει αντιστάθμιση ενός ισχυρά παραβιαζόμενου περιορισμού, με έναν περιορισμό που εμφανίζει μικρότερη παραβίαση. Αυτό είναι το ζήτημα που απασχολεί και δυσχεραστεί τον υπεύθυνο κατάρτισης του ωρολογίου προγράμματος.

Τα παραπάνω χαρακτηριστικά που εμφανίζονται κατά την επίλυση ενός προβλήματος σχεδιασμού ωρολογίου προγράμματος, και η ανάγκη αντιμετώπισής τους, οδήγησαν τους ερευνητές στην ανάπτυξη της εφαρμογής της πολυκριτηριακής ανάλυσης. Στόχος τους είναι να προσφέρουν έναν πιο ευέλικτο τρόπο διαχείρισης και χειρισμούς των διαφορετικών τύπων περιορισμών ταυτόχρονα.

Για το λόγο αυτό, επειδή τα τελευταία χρόνια η πλειονότητα των προβλημάτων και τον αλγορίθμων που χρησιμοποιούνται για την επίλυσή τους στηρίζεται στην πολυκριτηριακή ανάλυση, στη συνέχεια του κεφαλαίου, έχει επιλεγεί να αναλυθεί αλλά και να εφαρμοστεί σε δεδομένα της σειράς προβλημάτων Beasley, μία προσέγγιση της πολυκριτηριακής ανάλυσης βασισμένη στον **συναινετικό προγραμματισμό (Compromise Programming)**

καθεμίας χωριστά αντικειμενικής συνάρτησης $g_i(x), i=1,2,...,n$ και την αντικατάσταση της εκάστοτε βέλτιστης λύσης στις υπόλοιπες αντικειμενικές συναρτήσεις.

Τύπος λύσης	g_1 g_2 ... g_i ... g_n	Αντιστοιχούσα λύση
$[\max]g_1(x)$	g_{1*} g_{12} ... g_{1i} ... g_{1n}	x_1^1 x_2^1 ... x_l^1
$[\max]g_2(x)$	g_{21} g_{2*} ... g_{2i} ... g_{2n}	x_1^2 x_2^2 ... x_l^2
...	...	...
$[\max]g_i(x)$	g_{i1} g_{i2} ... g_{i*} ... g_{in}	x_1^i x_2^i ... x_l^i
...	...	...
$[\max]g_n(x)$	g_{n1} g_{n2} ... g_{ni} ... g_{n*}	x_1^n x_2^n ... x_l^n

Σχήμα 2: Πίνακας Πληρωμών

Συνεπώς, μετά τη μεγιστοποίηση της αντικειμενικής συνάρτησης $g_i(x)$, ονομάζουμε x^l την επιτευχθείσα βέλτιστη λύση και έχουμε:

$$g_i^* = \max_{x \in A} g_i(x)$$

$$g_{ij} = g_j(x^l), j \neq i$$

Ο πίνακας πληρωμών μας παρέχει πολύ χρήσιμες πληροφορίες, τόσο για την ποιότητα των λύσεων στις οποίες οδηγούν οι αντικειμενικές συναρτήσεις, όσο και για τον ανταγωνισμό μεταξύ των λύσεων.

Αναφορικά με την πολυκριτηριακή ανάλυση, αξίζει να σημειωθούν οι παρακάτω ορισμοί 2 σημαντικών εννοιών:

- ❖ Κυριαρχία (dominance): Μία δυνατή λύση x λέγεται κυρίαρχη μιας άλλης y ($x \Delta y$) εάν και μόνο εάν ισχύει: $x \Delta y \Leftrightarrow g_i(x) \geq g_i(y), \forall i$ και για έναν τουλάχιστον δείκτη i^* : $g_{i^*}(x) > g_{i^*}(y)$.
- ❖ Αποτελεσματικότητα (efficiency): Μία δυνατή λύση $x \in A$ λέγεται αποτελεσματική για την οικογένεια κριτηρίων $\{g_1, g_2, ..., g_n\}$ εάν και μόνο εάν δεν υπάρχει δυνατή λύση $y \in A$ η οποία κυριαρχεί της x . Για να είναι μία δυνατή λύση $x \in A$ αποτελεσματική, αρκεί να υπάρχουν πραγματικοί θετικοί

συντελεστές $\lambda_1, \lambda_2, \dots, \lambda_n$ ώστε η x να μεγιστοποιεί τη σύνθετη αντικειμενική

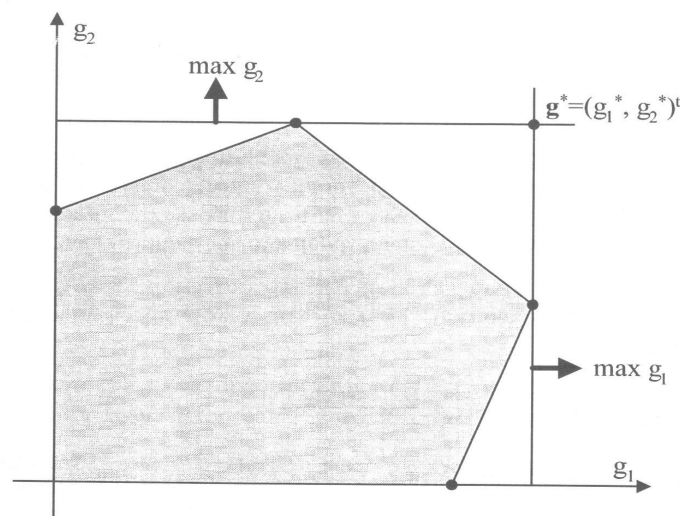
συνάρτηση $\sum_{i=1}^n \lambda_i g_i(x)$ πάνω στο σύνολο A .

Η πολυκριτηριακής ανάλυσης βασιζόμενη στον συναινετικό προγραμματισμό (Compromise Programming), είναι μία μεθοδολογία λήψης αποφάσεων που χρησιμοποιεί για την επίλυση, την απόσταση μεταξύ μιας λύσης και μίας ιδεώδους λύσης (ideal point) για τον προσδιορισμό και την αποτίμηση της ποιότητας της λύσης.

Ως ιδεώδης λύση (ideal point) ορίζεται μία λύση η οποία βελτιστοποιεί όλα τα κριτήρια ταυτόχρονα. Δηλαδή, είναι το διάνυσμα των τιμών των μεταβλητών απόφασης που μεγιστοποιεί ταυτόχρονα όλα τα κριτήρια και έχει συντεταγμένες στο χώρο των κριτηρίων:

$$g^* = (g_1^*, g_2^*, \dots, g_n^*)^t$$

Φυσικά δεν πρόκειται για δυνατή λύση, αφού το σημείο g^* βρίσκεται έξω από το επιτρεπτό σύνολο τιμών των g_i (Σχήμα 4).



Σχήμα 3: Χώρος κριτηρίων και ιδεώδης λύση

(ο σκιασμένος χώρος απεικονίζει το σύνολο των δυνατών λύσεων A)

Συγκεκριμένα για ένα πρόβλημα ωρολόγιου προγραμματισμού η ιδεώδης λύση είναι ένα ωρολόγιο πρόγραμμα που δεν παραβιάζει κανέναν περιορισμό. Όπως ήδη αναφέραμε, μία τέτοια λύση στην αληθινή ζωή, σε ένα πρόβλημα με πολλούς περιορισμούς δεν υπάρχει. Συναντάμε επίσης και την έννοια της αντι-ιδεώδους λύσης

(anti-ideal point) που χρησιμοποιείται για να δηλώσει ότι μία λύση δίνει την χειρότερη δυνατή τιμή σε όλα τα κριτήρια.

Μία ακόμα πολύ σημαντική έννοια είναι η συναινετική λύση. **Η συναινετική λύση (compromise solution) είναι μία δυνατή λύση, η οποία βρίσκεται σε ελάχιστη απόσταση από το ιδεώδες σημείο g^* .**

Ο ορισμός της συναινετικής λύσης προϋποθέτει την ύπαρξη κάποιας απόστασης των σημείων του χώρου A από το ιδεώδες σημείο (ideal point). Η απόσταση που συνήθως χρησιμοποιείται είναι η εξής:

$$F_p(x) = \left[\sum_{i=1}^n p_i \left(\frac{g_i^* - g_i(x)}{g_i^*} \right)^p \right]^{1/p},$$

όπου p είναι φυσικός αριθμός που επιλέγεται από του υπεύθυνο σχεδιασμού του ωρολόγιου προγράμματος και $p_1, p_2, \dots, p_n$ είναι συντελεστές βαρύτητας των κριτηρίων με άθροισμα την μονάδα ($\sum_i p_i = 1$).

Για να είναι εφικτή η έννοια της βελτιστοποίησης στο σημείο αυτό της επίλυσης του προβλήματος, είναι αναγκαίο να ικανοποιηθεί το υπερκριτήριο της ελαχιστοποίησης της απόστασης της συναινετικής λύσης (compromise solution) από την ιδεώδη λύση. Η βελτιστοποίηση αυτού του υπερκριτηρίου, αποτελεί πρόβλημα γραμμικού προγραμματισμού για τιμές της παραμέτρου P , $p=1$ και $p=\infty$. Σε διαφορετική περίπτωση, δηλαδή για τιμές της παραμέτρου P , $p \neq 1, \infty$, το πρόβλημά μας ανάγεται σε πρόβλημα μη γραμμικού προγραμματισμού.

Η απόσταση υπολογίζεται από τον εξής τύπο:

$$F_\infty(x) = \lim_{p \rightarrow \infty} F_p(x) = \max_{i=1,2,\dots,n} \left[p_i \frac{g_i^* - g_i(x)}{g_i^*} \right]$$

Για να προσδιορίσουμε την δυνατή λύση που ελαχιστοποιεί την απόσταση $F_\infty(x)$ αρκεί να υπολογίσουμε το παρακάτω ισοδύναμο του προβλήματος:

$$[\min] z = \lambda$$

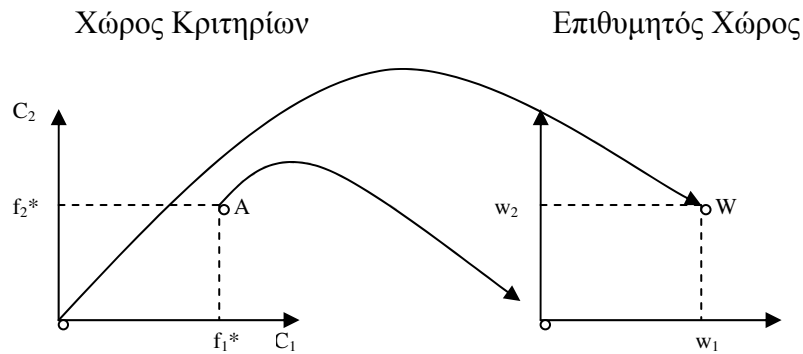
με τους περιορισμούς:

$$\lambda \geq \left[g_i^* - g_i(x) \right] \frac{p_i}{g_i^*}$$

$$\forall i = 1, 2, \dots, n$$

$$x \in A, \lambda \geq 0$$

Στη λήψη αποφάσεων βάσει της πολυκριτηριακής ανάλυσης, όπως έχει ήδη αναφερθεί, κάθε λύση εκφράζεται ως σημείο στον χώρο κριτηρίων (criteria space). Οι διαστάσεις του χώρου κριτηρίων είναι ίσες με τον αριθμό των κριτηρίων. Για να αντιμετωπιστεί το πρόβλημα της σύγκρισης διαφορετικής αξίας κριτηρίων, ένας νέος επιθυμητός χώρος (preference space) προτείνεται από τον Petrovic, 1995. Η γραμμική απεικόνιση που φαίνεται στο σχήμα 2 καθορίζει και αντιστοιχίζει τον χώρο κριτηρίων μέσα στον επιθυμητό χώρο. Στο συγκεκριμένο σχήμα υπάρχουν μόνο δύο κριτήρια C_1 και C_2 .



Σχήμα 4:

Για κάθε κριτήριο C_k $k = 1, \dots, K$ θέτουμε μία γραμμική συνάρτηση η οποία απεικονίζει την τιμή του κριτηρίου f_k ως τιμή $s_k \in [0, w_k]$, όπου w_k είναι η σχετική σημαντικότητα του κριτηρίου. Η γραμμική συνάρτηση (s_k) ορίζεται με τέτοιο τρόπο ώστε η χειρότερη τιμή του κριτηρίου, που δηλώνεται με f_k^* , σχεδιάζεται στο σημείο 0, και η καλύτερη τιμή του κριτηρίου, που δηλώνεται με f_k^* , σχεδιάζεται στο σημείο w_k . Έτσι:

$$s_k = w_k \frac{f_k^* - f_k}{f_k^* - f_k^*},$$

όπου τα f_k και s_k δηλώνουν τις τιμές των κριτηρίων C_k στον χώρο κριτηρίων (criteria space) και τον επιθυμητό χώρο (preference space) αντίστοιχα.

Σύμφωνα με την παραπάνω σχέση η ιδεώδης λύση (ideal point) $I = (f_1^*, \dots, f_k^*)$, $f_k^* = 0$, $k=1, \dots, K$ τοποθετείται στο σημείο $W = (w_1, \dots, w_k)$, ενώ η αντι-ιδεώδης λύση (anti-ideal point) τοποθετείται στο σημείο 0 του επιθυμητού χώρου (preference space). Συχνά, είναι αδύνατο να υπολογισθούν οι χειρότερες τιμές των κριτηρίων σε ένα σύνθετο πρόβλημα ωρολόγιου προγραμματισμού και ως εκ τούτου η αντι-ιδεώδης λύση (anti-ideal point) μπορεί να υπολογισθεί μόνο κατά προσέγγιση.

Για την ευρετική αναζήτηση (heuristic search) του επιθυμητού χώρου (preference space) αναπτύχθηκε ένας νέος αλγόριθμος, ο οποίος αποτελείται από δύο φάσεις.

1. Κατά τη διάρκεια της πρώτης φάσης στόχος του αλγορίθμου είναι να βρεθούν οι αρχικές λύσεις (initial solutions) που είναι αποδεκτά καλές για ένα μόνο κριτήριο. Η αναζήτηση αυτή επιτυγχάνεται με τη χρήση του Saturation degree ευρετικού, το οποίο αρχικά κατατάσσει τα μαθήματα και στη συνέχεια τοποθετεί τα μαθήματα στις χρονικές περιόδους του προγράμματος, με τρόπο ώστε να προκαλείται η μικρότερη δυνατή αύξηση στις τιμές των κριτηρίων.
2. Στη δεύτερη φάση, ο αλγόριθμος κατ' επανάληψη ερευνά το σύνολο τιμών (neighbor) της κάθε αρχικής λύσης (initial solution), με σκοπό να πλησιάσει στην ιδεώδη λύση (ideal point). Δύο τελεστές που χρησιμοποιούνται στη συγκεκριμένη αναζήτηση είναι οι ακόλουθοι:
 - a. Hill Climbing που στοχεύει στην αναζήτηση του τοπικού βέλτιστου (local optima). Η διαδικασία είναι η εξής: επιλέγει τυχαία ένα μάθημα και το επανασχεδιάζει στο ωρολόγιο πρόγραμμα, τοποθετώντας το σε μία διαφορετική χρονική περίοδο, με την προϋπόθεση ότι δεν προκαλείται παραβίαση κανενός σκληρού περιορισμού (hard constraint). Το ζητούμενο, με τον επαναπρογραμματισμό του μαθήματος είναι να επιτευχθεί η μέγιστη δυνατή μείωση στην απόσταση της λύσης μας από την ιδεώδη λύση (ideal point).
 - b. Ένας τελεστής μεταβολής (mutation operator) που διερευνά, πέρα από το τοπικό βέλτιστο (local optima), την νέα περιοχή του επιθυμητού χώρου. Επανασχεδιάζει τα μαθήματα που συμβάλλουν περισσότερο στην απόσταση από την ιδεώδη λύση (ideal point).

Η αναζήτηση διακόπτεται όταν οι πολλαπλές εφαρμογές του τελεστή Hill Climbing, και κατ' ακολουθία η εφαρμογή του τελεστή μεταβολής (mutation operator) δεν μειώνουν την απόσταση της λύσης μας από την ιδεώδη λύση (ideal point) για ένα προκαθορισμένο αριθμό βημάτων-επαναλήψεων. Επομένως, η αναζήτηση των συνόλων τιμών για κάθε αρχική λύση καταλήγει σε ένα ωρολόγιο πρόγραμμα (timetable). Το τελικό ωρολόγιο πρόγραμμα είναι η λύση που βρίσκεται πιο κοντά στην ιδεώδη λύση (ideal point).

Για την αξιολόγηση και τον υπολογισμό της απόστασης μεταξύ της λύσης $S=(s_1,s_2,\dots,s_k)$ και της ιδεώδους λύσης $W=(w_1,w_2,\dots,w_k)$, στον επιθυμητό χώρο, χρησιμοποιείται το σύνολο των L_p , όπου:

$$L_p(S,W) = \left\{ \sum_{k=1}^K [s_k - w_k]^p \right\}^{1/p} \text{ με } 1 \leq p \leq \infty$$

Η τιμή της παραμέτρου p επιτρέπει στον υπεύθυνο του σχεδιασμού του ωρολογίου προγράμματος να εκφράσει την προτίμησή του απέναντι στις λύσεις. Τρεις τιμές της παραμέτρου p παρουσιάζουν συνήθως ιδιαίτερο ενδιαφέρον.

- ❖ $p=1$: Αυτή η τιμή της παραμέτρου επιτρέπει την ικανοποίηση κάποιων περιορισμών στο ωρολόγιο πρόγραμμα σε πολύ υψηλό βαθμό, αντισταθμίζοντας την χαμηλή ικανοποίηση κάποιων άλλων περιορισμών.
- ❖ $p=\infty$: (το οποίο προκύπτει από $L_\infty(S,W) = \max_k [w_k - s_k]$) Όταν η παράμετρος p έχει αυτή την τιμή, προλαμβάνει και αποτρέπει τις εναλλαγές μεταξύ των τιμών των κριτηρίων. Για παράδειγμα, μόνο οι λύσεις που ικανοποιούν όλους τους περιορισμούς σε αρκετά ικανοποιητικό βαθμό, έχουν ενδιαφέρον.
- ❖ $p=2$: Η τιμή 2, μας αποδίδει την λύση που βρίσκεται γεωμετρικά πλησιέστερα στην ιδεώδη λύση στον επιθυμητό χώρο.

Η προσέγγιση επίλυσης με τη βοήθεια της πολυκριτηριακής ανάλυσης βασιζόμενη στον συναινετικό προγραμματισμό (Compromise Programming), είναι μία μεθοδολογία που δεν μπορεί να συγκριθεί με οποιαδήποτε άλλη μέθοδο επίλυσης που χρησιμοποιεί μία μοναδική συνάρτηση κόστους (αντικειμενική συνάρτηση).

Αιτία του γεγονότος αυτού είναι ότι τα κριτήρια, οι μετρητές που χρησιμοποιούνται για την αξιολόγηση του ωρολογίου προγράμματος (timetable) είναι μη συγκρίσιμα και εφαρμόζονται με διαφορετικούς περιορισμούς. Ωστόσο, η προσέγγιση αυτή, παρέχει μεγαλύτερη ευελιξία στον υπεύθυνο σχεδιασμού του ωρολογίου προγράμματος, στην έκφραση της προτίμησής του. Η δυνατότητα αυτή προσφέρεται στον υπεύθυνο του σχεδιασμού καθώς του εκχωρείται η σχετική σημαντικότητα των κριτηρίων (δεν χρειάζεται να ανησυχεί εάν είναι εύκολη η ικανοποίηση των αντίστοιχων περιορισμών). Επίσης, η συγκεκριμένη προσέγγιση επιτρέπει στον σχεδιαστή του ωρολογίου προγράμματος να επιλέξει τον τρόπο υπολογισμού της απόστασης, αντικατοπτρίζοντας έτσι την θέση σχετικά με την αντιστάθμιση μεταξύ των διαφόρων τιμών των κριτηρίων.

4.3 Επίλυση 1^{ου} προβλήματος

4.3.1 Δεδομένα και απαιτήσεις του προβλήματος

Τα δεδομένα και οι απαιτήσεις του προβλήματος είναι τα στοιχεία που θα μας υποδείξουν τον τρόπο με τον οποίο θα αναπτύξουμε τη μοντελοποίηση του προβλήματος. Στο συγκεκριμένο παράδειγμα που επιλύουμε τα δεδομένα είναι:

Αριθμός Καθηγητών (Teachers)	4
Αριθμός Μαθημάτων (Subjects)	30
Αριθμός Τμημάτων Σπουδαστών (Classes)	4
Αριθμός Διαθέσιμων Αιθουσών (Room Available)	4
Αριθμός Απαιτήσεων (Requirements)	120

Πίνακας 4.1: Δεδομένα

Πρέπει να σημειωθεί ότι μας δίνεται σαν δεδομένο το στοιχείο ότι το ωρολόγιο πρόγραμμα αναλύεται σε 6 χρονικές περιόδους την ημέρα, 5 ημέρες την εβδομάδα.

Οι απαιτήσεις του προβλήματος, που όπως δηλώνεται στον Πίνακα 4.1 είναι 120, αναλύονται στον πίνακα 4.2. Συγκεκριμένα τα χρωματιστά κελιά δηλώνουν

πόσες φορές θα συναντήσει ο κάθε καθηγητής, το κάθε τμήμα, σε κάθε αίθουσα. Για παράδειγμα το κελί με αριθμό 6 στην 3^η γραμμή και 4^η στήλη, δηλώνει ότι ο καθηγητής 4, πρέπει να συναντήσει το τμήμα 3, 6 φορές στην αίθουσα 1.

Απαιτήσεις q

Τμήμα i	Καθηγητής j				
	1	2	3	4	
1	2	2	1	2	1 k
2	1	1	1	2	
3	1	1	1	6	
4	2	2	3	2	
1	2	5	1	2	2 k
2	0	4	3	2	
3	1	2	1	0	
4	2	2	1	2	
1	2	1	1	2	3 k
2	0	0	5	1	
3	2	1	4	1	
4	6	1	2	1	
1	3	1	2	1	4 k
2	1	4	1	4	
3	3	3	2	1	
4	2	0	1	1	

Πίνακας 4.2: Απαιτήσεις q

Οι επιπλέον πληροφορίες που έχουμε για το πρόβλημα αφορούν στους συντελεστές βαρύτητας-σημαντικότητας, που στο παράδειγμά μας, με εξαίρεση το συντελεστή βαρύτητα της διανομής που είναι 0, οι υπόλοιποι είναι όλοι ίσοι με 1.

Importance of of Classclashes	1
Importance of Teacherclashes	1
Importance of Roomclashes	1
Importance of Blocks	1
Importance of preference	1
Importance of limits	1
Importance of relations	1
Importance of distribution	0

Πίνακας 4.3: Συντελεστές Σημαντικότητας

4.3.2 Μοντελοποίηση του 1^{ου} προβλήματος

Σύμφωνα με τα δεδομένα και τις απαιτήσεις του προβλήματος η μοντελοποίηση είναι η ακόλουθη:

↪ Αρχικά ορίζουμε τη **μεταβλητή** του προβλήματος, η οποία είναι η εξής:

$$x_{ijkt}, \text{ όπου}$$

i	Τμήμα	4
j	Καθηγητής	4
k	Αίθουσα	4
t	Χρονική Περίοδος	30

↪ Στη συνέχεια θα διαμορφώσουμε σε πίνακα, με βάση την αίθουσα το πρόγραμμα, το οποίο θα περιλαμβάνει όλες τις παραπάνω μεταβλητές x_{ijkt} :

Αίθουσα k																
	Καθηγητής 1 j				Καθηγητής 2 j				Καθηγητής 3 j				Καθηγητής 4 j			
	t ₁	t ₂	...	t ₃₀	t ₁	t ₂	...	t ₃₀	t ₁	t ₂	...	t ₃₀	t ₁	t ₂	...	t ₃₀
Τμήμα 1 i																
Τμήμα 2 i																
Τμήμα 3 i																
Τμήμα 4 i																

↪ Για να μπορέσουμε όμως να ορίσουμε τις αντικειμενικές μας συναρτήσεις θα πρέπει να προσθέσουμε κάποιες επιπλέον μεταβλητές, που μας εξασφαλίζουν ότι δεν θα προκύψουν επικαλύψεις είτε στους Καθηγητές, είτε στα Τμήματα. Με άλλα λόγια, θα ικανοποιούν τους σκληρούς περιορισμούς του προβλήματος που είναι:

- 1) Να μην τοποθετηθεί ο ίδιος καθηγητής, την ίδια χρονική περίοδο σε παραπάνω από ένα τμήματα.
- 2) Να μην οριστούν περισσότερα από ένα μαθήματα, σε ένα τμήμα την ίδια χρονική περίοδο.

Οι μεταβλητές που θα ορίσουμε είναι:

1) για τους καθηγητές j , οι d_j^+ και d_j^- , όπου ισχύει $\sum x_{ijkt} - d_j^+ + d_j^- = 1$ για κάθε i, k, t

2) για τα τμήματα i , οι d_i^+ και d_i^- , όπου ισχύει $\sum x_{ijkt} - d_i^+ + d_i^- = 1$ για κάθε j, k, t

Οι αντικειμενικές συναρτήσεις, θα είναι δύο και θα είναι συναρτήσεις ελαχιστοποίησης.

Ψ Έτσι για τους καθηγητές θα έχουμε αντικειμενική συνάρτηση την:

$$\min(\sum d_j^+),$$

υπό τους περιορισμούς:

$$x_{ijkt} = binary$$

$$d_j^+ \geq 0 \text{ και } d_j^- \geq 0$$

$$\sum_{j=1}^4 x_{ijkt} d_j^+ + d_j^- = 1$$

$$\sum x_{ijkt} \leq 1 \text{ για κάθε } i, k, t$$

Για την ικανοποίηση των απαιτήσεων πρέπει να ισχύει για κάθε $\sum_{k=1}^4 x_{ijkt} = q_{ijkt}$ όπου q

ο αριθμός των απαιτήσεων όπως προκύπτει από τον πίνακα 4.2.

Ψ Κατ' αντιστοιχία για τα τμήματα των σπουδαστών θα έχουμε αντικειμενική συνάρτηση την:

$$\min(\sum d_i^+),$$

υπό τους περιορισμούς:

$$x_{ijkt} = binary$$

$$d_i^+ \geq 0 \text{ και } d_i^- \geq 0$$

$$\sum_{i=1}^4 x_{ijkt} - d_i^+ + d_i^- = 1$$

$$\sum x_{ijkt} \leq 1 \text{ για κάθε } i, k, t$$

Για την ικανοποίηση των απαιτήσεων πρέπει να ισχύει για κάθε $\sum_{k=1}^4 x_{ijkt} = q_{ijkt}$ όπου q ο αριθμός των απαιτήσεων όπως προκύπτει από τον πίνακα 4.2.

4.3.3 Επίλυση αντικειμενικής συνάρτησης Καθηγητών $\min(\sum d_j^+)$

Επιλύοντας το γραμμικό πρόβλημα στον H/Y, με τη βοήθεια του Premium Solver Platform προκύπτει η ακόλουθη λύση:

$$\sum d_j^+ = 0$$

Το ωρολόγιο πρόγραμμα που προέκυψε κατά την επίλυση της 1^{ης} αντικειμενικής μας συνάρτησης $\min(\sum d_j^+)$ παρουσιάζεται στους παρακάτω πίνακες:

		Teacher 1																													
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30
ROOM 1	Class 1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
ROOM 2	Class 1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0
ROOM 3	Class 1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	1	0	1	1	0	0	0	0
ROOM 4	Class 1	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
	Class 4	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

		Teacher 2																														
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30	
ROOM 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
ROOM 2	Class 1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	1	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
ROOM 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ROOM 4	Class 1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0
	Class 3	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

		Teacher 3																														
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30	
ROOM 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0
ROOM 2	Class 1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
ROOM 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0
	Class 4	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
ROOM 4	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	
	Class 2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

		Teacher 4																														
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30	
ROOM 1	Class 1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	
	Class 3	0	0	0	0	0	0	1	1	1	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
ROOM 2	Class 1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0
ROOM 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ROOM 4	Class 1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0
	Class 3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

4.3.4 Επίλυση αντικειμενικής συνάρτησης Τμημάτων $\min(\sum d_i^+)$

Επιλύοντας το γραμμικό πρόβλημα στον H/Y, με τη βοήθεια του Premium Solver Platform προκύπτει η ακόλουθη λύση:

$$\sum d_i^+ = 0$$

Το ωρολόγιο πρόγραμμα που προέκυψε κατά την επίλυση της 2^{ης} αντικειμενικής μας συνάρτησης $\min(\sum d_i^+)$ παρουσιάζεται στους παρακάτω πίνακες:

		Teacher 1																														
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30	
ROOM 1	Class 1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
ROOM 2	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
ROOM 3	Class 1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	1	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
ROOM 4	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0

		Teacher 2																													
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30
ROOM 1	Class 1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ROOM 2	Class 1	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0
ROOM 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
ROOM 4	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

		Teacher 3																														
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30	
ROOM 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
ROOM 2	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	Class 2	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
	Class 4	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ROOM 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1	0	0	0
	Class 4	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
ROOM 4	Class 1	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

		Teacher 4																														
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30	
ROOM 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	1	1	1	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ROOM 2	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
	Class 2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
ROOM 3	Class 1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ROOM 4	Class 1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

☞ Και στις δύο περιπτώσεις επίλυσης του προβλήματος, βρέθηκε λύση που βελτιστοποιεί τις αντικειμενικές μας συναρτήσεις και να ικανοποιεί όλους τους περιορισμούς

☞ Αξίζει να σημειωθεί ότι κατά την επίλυση και των δύο αντικειμενικών συναρτήσεων οι τιμές των μεταβλητών οι d_j^+ και d_j^- για τους καθηγητές, και d_i^+ και d_i^- για τα τμήματα των σπουδαστών έλαβαν τιμή 0.

4.3.5 Ελαχιστοποίηση της απόστασης – Τελικό Ωρολόγιο Πρόγραμμα

Το τελευταίο μέρος της επίλυσης του προβλήματος θα μας οδηγήσει και στην κατάρτιση του τελικού Ωρολογίου Προγράμματος. Στο σημείο αυτό θα πρέπει να λύσουμε ξανά το πρόβλημα, το οποίο θα λάβει την εξής μοντελοποίηση, ώστε να γίνει εφικτή η ελαχιστοποίηση της απόστασης της λύσης μας από την βέλτιστη λύση. Πιο συγκεκριμένα, η αντικειμενική μας συνάρτηση θα είναι η συνάρτηση ελαχιστοποίησης:

$$\min z = \lambda$$

υπό τους περιορισμούς:

$$x_{ijkt} = binary$$

$$d_j^+ \geq 0 \text{ και } d_j^- \geq 0, d_i^+ \geq 0 \text{ και } d_i^- \geq 0$$

$$\sum_{j=1}^4 x_{ijkt} d_j^+ + d_j^- = 1, \sum_{i=1}^4 x_{ijkt} -d_i^+ + d_i^- = 1$$

$$\sum x_{ijkt} \leq 1 \text{ για κάθε } i,k,t$$

☞ Για την ικανοποίηση των απαιτήσεων πρέπει να ισχύει για κάθε $\sum_{k=1}^4 x_{ijkt} = q_{ijkt}$ όπου q ο

αριθμός των απαιτήσεων όπως προκύπτει από τον πίνακα 4.2.

☞ Επιπλέον από τον Πίνακα Πληρωμών που ακολουθεί προκύπτουν δύο ακόμη περιορισμοί:

$$\lambda \geq 40$$

$$\lambda \geq 43$$

ΠΙΝΑΚΑΣ ΠΛΗΡΩΜΩΝ		
	$g1(x)_{teacher}$	$g2(x)_{class}$
$\min [g1(x)_{teacher}]$	0	43
$\min [g2(x)_{class}]$	40	0

Όπου $\min [g1(x)_{teacher}] = \min(\sum d_j^+)$ και $\min [g2(x)_{class}] = \min(\sum d_i^+)$

↪ Στις μεταβλητές του προβλήματος εκτός από τις $x_{ijk t}$, d_j^+ , d_j^- , d_i^+ και d_i^- , θα ορίσουμε επίσης και τη μεταβλητή λ .

Επιλύοντας το γραμμικό πρόβλημα στον H/Y, με τη βοήθεια του Premium Solver Platform προκύπτει η ακόλουθη λύση:

$$\min z = 43, \text{ αφού } \lambda = 43$$

Το τελικό πια, ωρολόγιο πρόγραμμα που προέκυψε παρουσιάζεται στους παρακάτω πίνακες:

		Teacher 1																														
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30	
ROOM 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
ROOM 2	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ROOM 3	Class 1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	1	0	1	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
ROOM 4	Class 1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0

		Teacher 2																																
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30			
ROOM 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0		
	Class 2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0		
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0		
ROOM 2	Class 1	0	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0		
	Class 2	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	
ROOM 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
ROOM 4	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

		Teacher 3																													
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30
ROOM 1	Class 1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ROOM 2	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
ROOM 3	Class 1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	1	0	0	0	0
	Class 3	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
ROOM 4	Class 1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
	Class 4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

		Teacher 4																														
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30	
ROOM 1	Class 1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
	Class 3	1	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0
ROOM 2	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
ROOM 3	Class 1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
ROOM 4	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

4.4 Επίλυση 2^{ου} προβλήματος

4.4.1 Δεδομένα και απαιτήσεις του προβλήματος

Τα δεδομένα και οι απαιτήσεις του προβλήματος είναι τα στοιχεία που θα μας υποδείξουν τον τρόπο με τον οποίο θα αναπτύξουμε τη μοντελοποίηση του προβλήματος. Στο συγκεκριμένο παράδειγμα που επιλύουμε τα δεδομένα είναι:

Αριθμός Καθηγητών (Teachers)	6
Αριθμός Μαθημάτων (Subjects)	30
Αριθμός Τμημάτων Σπουδαστών (Classes)	6
Αριθμός Διαθέσιμων Αιθουσών (Room Available)	6
Αριθμός Απαιτήσεων (Requirements)	180

Πίνακας 4.4: Δεδομένα

Πρέπει να σημειωθεί ότι μας δίνεται σαν δεδομένο το στοιχείο ότι το ωρολόγιο πρόγραμμα αναλύεται σε 6 χρονικές περιόδους την ημέρα, 5 ημέρες την εβδομάδα.

Οι απαιτήσεις του προβλήματος, που όπως δηλώνεται στον Πίνακα 4.4 είναι 180, αναλύονται στον πίνακα 4.5. Συγκεκριμένα τα χρωματιστά κελιά δηλώνουν πόσες φορές θα συναντήσει ο κάθε καθηγητής, το κάθε τμήμα, σε κάθε αίθουσα, όπως ακριβώς και κατά την επίλυση του 1ου προβλήματος.

Απαιτήσεις q

	Καθηγητής j						
Τμήμα i	1	2	3	4	5	6	
1	1	0	1	1	0	3	Αίθουσα 1 k
2	0	2	0	0	0	0	
3	0	1	0	1	0	0	
4	1	0	2	1	0	1	
5	2	1	1	2	2	1	
6	0	1	1	1	1	2	

30 t

1	2	1	1	0	2	0	Αίθουσα 2 k	30 t
2	0	0	1	0	2	3		
3	0	0	0	0	2	1		
4	1	0	0	0	0	0		
5	0	2	1	2	2	1		
6	3	0	1	1	1	0		
1	2	1	1	1	0	0	Αίθουσα 3 k	30 t
2	1	0	3	0	1	0		
3	0	0	1	1	2	1		
4	1	1	1	1	1	2		
5	1	1	1	0	0	0		
6	1	1	2	0	1	0		
1	0	2	0	0	2	1	Αίθουσα 4 k	30 t
2	0	1	1	2	0	2		
3	0	2	0	0	3	0		
4	1	1	2	1	0	0		
5	1	0	1	0	1	2		
6	1	0	0	1	1	1		
1	0	1	0	0	0	3	Αίθουσα 5 k	30 t
2	0	1	0	3	1	0		
3	3	0	1	4	0	2		
4	2	0	1	0	0	1		
5	2	1	0	0	0	0		
6	1	1	0	0	1	1		
1	0	1	1	0	2	0	Αίθουσα 6 k	30 t
2	0	3	2	1	0	0		
3	0	1	0	2	1	1		
4	1	2	2	3	0	0		
5	1	0	0	0	1	0		
6	1	1	1	1	0	1		

Πίνακας 4.5: Απαιτήσεις q

Οι επιπλέον πληροφορίες που έχουμε για το πρόβλημα αφορούν στους συντελεστές βαρύτητας-σημαντικότητας, που στο παράδειγμά μας, με εξαίρεση το συντελεστή βαρύτητα της διανομής που είναι 0, οι υπόλοιποι είναι όλοι ίσοι με 1.

Importance of of Classclashes	1
Importance of Teacherclashes	1
Importance of Roomclashes	1
Importance of Blocks	1
Importance of preference	1
Importance of limits	1
Importance of relations	1
Importance of distribution	0

Πίνακας 4.6: Συντελεστές Σημαντικότητας

4.4.2 Μοντελοποίηση 2^{ου} προβλήματος

Σύμφωνα με τα δεδομένα και τις απαιτήσεις του προβλήματος η μοντελοποίηση είναι η ακόλουθη:

↪ Αρχικά ορίζουμε τη βασική **μεταβλητή** του προβλήματος, η οποία είναι η εξής:

$$x_{ijkt} \text{ , όπου}$$

i	Τμήμα	6
j	Καθηγητής	6
k	Αίθουσα	6
t	Χρονική Περίοδος	30

↪ Στη συνέχεια θα διαμορφώσουμε σε πίνακα, με βάση την αίθουσα το πρόγραμμα, το οποίο θα περιλαμβάνει όλες τις παραπάνω μεταβλητές x_{ijkt} :

Αίθουσα k																								
	Καθηγητής 1 j				Καθηγητής 2 j				Καθηγητής 3 j				Καθηγητής 4 j				Καθηγητής 5 j				Καθηγητής 6 j			
	t ₁	t ₂	...	t ₃₀	t ₁	t ₂	...	t ₃₀	t ₁	t ₂	...	t ₃₀	t ₁	t ₂	...	t ₃₀	t ₁	t ₂	...	t ₃₀	t ₁	t ₂	...	t ₃₀
Τμήμα 1 i																								
Τμήμα 2 i																								
Τμήμα 3 i																								
Τμήμα 4 i																								
Τμήμα 5 i																								
Τμήμα 6 i																								

↪ Για να μπορέσουμε όμως να ορίσουμε τις αντικειμενικές μας συναρτήσεις θα πρέπει να προσθέσουμε κάποιες επιπλέον μεταβλητές, που μας εξασφαλίζουν ότι δεν θα προκύψουν επικαλύψεις είτε στους Καθηγητές, είτε στα Τμήματα. Με άλλα λόγια, θα ικανοποιούν τους σκληρούς περιορισμούς του προβλήματος που είναι:

- 3) Να μην τοποθετηθεί ο ίδιος καθηγητής, την ίδια χρονική περίοδο σε παραπάνω από ένα τμήματα.
- 4) Να μην οριστούν περισσότερα από ένα μαθήματα, σε ένα τμήμα την ίδια χρονική περίοδο.

Οι **μεταβλητές** που θα ορίσουμε είναι:

- 1) για τους καθηγητές j, οι d_j^+ και d_j^- , όπου ισχύει $\sum x_{ijkt} - d_j^+ + d_j^- = 1$ για κάθε i,k,t
- 2) για τα τμήματα i, οι d_i^+ και d_i^- , όπου ισχύει $\sum x_{ijkt} - d_i^+ + d_i^- = 1$ για κάθε j,k,t

Οι **αντικειμενικές συναρτήσεις**, θα είναι δύο και θα είναι συναρτήσεις **ελαχιστοποίησης**.

↪ Έτσι για τους καθηγητές θα έχουμε αντικειμενική συνάρτηση την:

$$\min(\sum d_j^+),$$

υπό τους περιορισμούς:

$$x_{ijkt} = binary$$

$$d_j^+ \geq 0 \text{ και } d_j^- \geq 0$$

$$\sum_{j=1}^6 x_{ijkt} d_j^+ + d_j^- = 1$$

$$\sum x_{ijkt} \leq 1 \text{ για κάθε } i, k, t$$

Για την ικανοποίηση των απαιτήσεων πρέπει να ισχύει για κάθε $\sum_{k=1}^6 x_{ijkt} = q_{ijkt}$ όπου q ο αριθμός των απαιτήσεων όπως προκύπτει από τον πίνακα 4.5.

↪ Κατ' αντιστοιχία για τα τμήματα των σπουδαστών θα έχουμε αντικειμενική συνάρτηση την:

$$\min(\sum d_i^+),$$

υπό τους περιορισμούς:

$$x_{ijkt} = \text{binary}$$

$$d_i^+ \geq 0 \text{ και } d_i^- \geq 0$$

$$\sum_{i=1}^6 x_{ijkt} - d_i^+ + d_i^- = 1$$

$$\sum x_{ijkt} \leq 1 \text{ για κάθε } i, k, t$$

Για την ικανοποίηση των απαιτήσεων πρέπει να ισχύει για κάθε $\sum_{k=1}^6 x_{ijkt} = q_{ijkt}$ όπου q ο αριθμός των απαιτήσεων όπως προκύπτει από τον πίνακα 4.5.

4.4.3 Επίλυση αντικειμενικής συνάρτησης Καθηγητών $\min(\sum d_j^+)$

Επιλύοντας το γραμμικό πρόβλημα στον H/Y, με τη βοήθεια του Premium Solver Platform προκύπτει η ακόλουθη λύση:

$$\sum d_j^+ = 0$$

Το ωρολόγιο πρόγραμμα που προέκυψε κατά την επίλυση της 1^{ης} αντικειμενικής μας συνάρτησης $\min(\sum d_j^+)$ παρουσιάζεται στους παρακάτω πίνακες:

		Teacher 1																															
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30		
Room 1	Class 1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 2	Class 1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
Room 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 4	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 5	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
	Class 4	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	
Room 6	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

		Teacher 3																															
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30		
Room 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	
	Class 5	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	
Room 2	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	
Room 3	Class 1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	
Room 4	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Room 5	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Room 6	Class 1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0		
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 6	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

[illegible]

		Teacher 5																														
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30	
Room 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 5	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
Room 2	Class 1	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 4	Class 1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Room 5	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 6	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		

		Teacher 6																													
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30
Room 1	Class 1	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
Room 2	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0
	Class 3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 4	Class 1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
Room 5	Class 1	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Room 6	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0

4.4.4 Επίλυση αντικειμενικής συνάρτησης Τμημάτων $\min(\sum d_i^+)$

Επιλύοντας το γραμμικό πρόβλημα στον H/Y, με τη βοήθεια του Premium Solver Platform προκύπτει η ακόλουθη λύση:

$$\sum d_i^+ = 0$$

Το ωρολόγιο πρόγραμμα που προέκυψε κατά την επίλυση της 2^{ης} αντικειμενικής μας συνάρτησης $\min(\sum d_i^+)$ παρουσιάζεται στους παρακάτω πίνακες:

		Teacher 1																													
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30
Room 1	Class 1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 2	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
Room 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 4	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 5	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0
	Class 4	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0
	Class 6	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 6	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

		Teacher 3																													
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30
Room 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Room 2	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 4	Class 1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 5	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 6	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

		Teacher 3																															
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30		
Room 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	
	Class 5	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 2	Class 1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
Room 3	Class 1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 4	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 5	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 6	Class 1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[illegible]

		Teacher 5																											
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
	Class 5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Room 2	Class 1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
	Class 2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 3	Class 1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Room 4	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 5	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Room 6	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

		Teacher 6																												
		0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Room 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 2	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
Room 4	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
Room 5	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 6	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

☞ Και στις δύο περιπτώσεις επίλυσης του προβλήματος, βρέθηκε λύση που βελτιστοποιεί τις αντικειμενικές μας συναρτήσεις και να ικανοποιεί όλους τους περιορισμούς

☞ Αξίζει να σημειωθεί ότι κατά την επίλυση και των δύο αντικειμενικών συναρτήσεων οι τιμές των μεταβλητών οι d_j^+ και d_j^- για τους καθηγητές, και d_i^+ και d_i^- για τα τμήματα των σπουδαστών έλαβαν τιμή 0.

4.4.5 Ελαχιστοποίηση της απόστασης – Τελικό Ωρολόγιο Πρόγραμμα

Το τελευταίο μέρος της επίλυσης του προβλήματος θα μας οδηγήσει και στην κατάρτιση του τελικού Ωρολογίου Προγράμματος. Στο σημείο αυτό θα πρέπει να λύσουμε ξανά το πρόβλημα, το οποίο θα λάβει την εξής μοντελοποίηση, ώστε να γίνει εφικτή η ελαχιστοποίηση της απόστασης της λύσης μας από την βέλτιστη λύση. Πιο συγκεκριμένα, η αντικειμενική μας συνάρτηση θα είναι η συνάρτηση ελαχιστοποίησης:

$$\min z = \lambda$$

υπό τους περιορισμούς:

$$x_{ijkt} = binary$$

$$d_j^+ \geq 0 \text{ και } d_j^- \geq 0, d_i^+ \geq 0 \text{ και } d_i^- \geq 0$$

$$\sum_{j=1}^4 x_{ijkt} d_j^+ + d_j^- = 1, \sum_{i=1}^4 x_{ijkt} -d_i^+ + d_i^- = 1$$

$$\sum x_{ijkt} \leq 1 \text{ για κάθε } i,k,t$$

☞ Για την ικανοποίηση των απαιτήσεων πρέπει να ισχύει για κάθε $\sum_{k=1}^4 x_{ijkt} = q_{ijkt}$ όπου q ο

αριθμός των απαιτήσεων όπως προκύπτει από τον πίνακα 4.4.

☞ Επιπλέον από τον Πίνακα Πληρωμών που ακολουθεί προκύπτουν δύο ακόμη περιορισμοί:

$$\lambda \geq 67$$

$$\lambda \geq 74$$

ΠΙΝΑΚΑΣ ΠΛΗΡΩΜΩΝ		
	$g1(x)_{teacher}$	$g2(x)_{class}$
$\min [g1(x)_{teacher}]$	0	67
$\min [g2(x)_{class}]$	74	0

Όπου $\min [g1(x)_{teacher}] = \min(\sum d_j^+)$ και $\min [g2(x)_{class}] = \min(\sum d_i^+)$

↪ Στις μεταβλητές του προβλήματος εκτός από τις $x_{ijk t}$, d_j^+ , d_j^- , d_i^+ και d_i^- , θα ορίσουμε επίσης και τη μεταβλητή λ .

Επιλύοντας το γραμμικό πρόβλημα στον H/Y, με τη βοήθεια του Premium Solver Platform προκύπτει η ακόλουθη λύση:

$$\min z = 74, \text{ αφού } \lambda = 74$$

Το τελικό πια, ωρολόγιο πρόγραμμα που προέκυψε παρουσιάζεται στους παρακάτω πίνακες:

		Teacher 1																													
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30
Room 1	Class 1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 2	Class 1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
Room 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 4	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 5	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1
	Class 4	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Room 6	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

		Teacher 2																														
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30	
Room 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	
Room 2	Class 1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Room 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 4	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
	Class 2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 5	Class 1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 6	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

		Teacher 3																														
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30	
Room 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	
Room 2	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Room 3	Class 1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 4	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 5	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 6	Class 1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

		Teacher 4																																
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30			
Room 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0		
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0		
	Class 4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0		
	Class 6	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Room 2	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 5	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Room 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Room 4	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
Room 5	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Room 6	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	
	Class 4	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 6	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

		Teacher 5																														
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30	
Room 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 5	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 2	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
	Class 2	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
Room 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 4	Class 1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
Room 5	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
Room 6	Class 1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

		Teacher 6																															
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21	T22	T23	T24	T25	T26	T27	T28	T29	T30		
Room 1	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0		
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0		
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	
Room 2	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 5	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Room 3	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 4	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 5	Class 1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Room 6	Class 1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Class 6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

5ο Κεφάλαιο

Συμπεράσματα

Τα συμπεράσματα που προκύπτουν από την εκπόνηση της παρούσης διπλωματικής εργασίας, είναι διττά. Από τη μία πλευρά παρουσιάσθηκε η σπουδαιότητα της ποιοτικής σχεδίασης ενός ωρολόγιου προγράμματος, το οποίο να ικανοποιεί όσο το δυνατόν περισσότερους μαλακούς περιορισμούς. Η ύπαρξης ενός ποιοτικού προγράμματος σε έναν οργανισμό ή εκπαιδευτικό ίδρυμα, είναι ζωτικής σημασία καθώς εξασφαλίζει την αποτελεσματικότερη και εύρυθμη λειτουργία του. Την σημαντικότητα του ωρολόγιου προγράμματος, μπορεί κάποιος να την αντιληφθεί εάν αναλογιστεί το πλήθος των ατόμων αλλά και των διαδικασιών που εμπλέκονται και επηρεάζεται η καθημερινότητά τους από το συγκεκριμένο πρόγραμμα.

Από την άλλη πλευρά, τίθεται το ζήτημα της επίλυσης του προβλήματος που προκύπτει με το σχεδιασμό του ωρολόγιου προγράμματος ενός εκπαιδευτικού ιδρύματος. Έτσι, ενώ κατανοούμε τη σημασία σχεδιασμού του ωρολόγιου προγράμματος, για τη βελτίωση της λειτουργίας των ιδρυμάτων, βρισκόμαστε αντιμέτωποι με ένα πολύπλοκο ζήτημα, του οποίου η επίλυση δεν είναι καθόλου εύκολη υπόθεση. Επιπλέον, η βιβλιογραφική ανασκόπηση έδειξε ότι οι μέθοδοι που έχουν προταθεί είναι πολλές, και συχνά βασίζονται σε εντελώς διαφορετικά επιστημονικά πεδία. Η δυσκολία επίλυσης του προβλήματος, έγκειται στην πολυπλοκότητά του, λόγω της ύπαρξης πολύ μεγάλου αριθμού μεταβλητών, αλλά και του ορισμού σημαντικού αριθμού περιορισμών. Η επιλογή της μεθόδου επίλυσης σχετίζεται με τη φύση και τα χαρακτηριστικά του ιδρύματος ή του οργανισμού, έτσι κάθε πρόβλημα είναι διαφορετικό και απαιτεί ξεχωριστό χειρισμό.

Οι σημαντικότερες και δημοφιλέστερες μέθοδοι-τεχνικές επίλυσης αναλύθηκαν στο Κεφάλαιο 3. Από τις παραπάνω μεθόδους κρίθηκε σκόπιμο να εφαρμοστεί σε δεδομένα της σειράς προβλημάτων Beasley, η πολυκριτηριακή ανάλυση με βάση τον συναινετικό προγραμματισμό. Η επιλογή της στηρίχθηκε στο γεγονός ότι η πολυκριτηριακή ανάλυση αποτελεί μία πολύ αποτελεσματική, εύχρηστη και ευρέως διαδεδομένη μεθοδολογία επίλυσης προβλημάτων χρονικού προγραμματισμού.

Πιο συγκεκριμένα, η πολυκριτηριακή ανάλυση βασισμένη στον συναινετικό προγραμματισμό, παρέχει μεγαλύτερη ευελιξία στον υπεύθυνο σχεδιασμού του ωρολόγιου προγράμματος. Το γεγονός αυτό συνίσταται στο ότι μπορεί να εκφράσει την προτίμησή του. Η δυνατότητα αυτή προσφέρεται στον υπεύθυνο του σχεδιασμού καθώς του εκχωρείται η δήλωση της σχετικής σημαντικότητας των κριτηρίων. Επιπλέον, ο συναινετικός προγραμματισμός επιτρέπει στον σχεδιαστή του ωρολόγιου προγράμματος να επιλέξει τον τρόπο υπολογισμού της απόστασης, καθορίζοντας έτσι την αντιστάθμιση μεταξύ των διαφόρων τιμών των κριτηρίων.

Ένα εξίσου σημαντικό κριτήριο, που αξίζει να αναφερθεί, και συνηγόρησε στην επιλογή της συγκεκριμένης μεθοδολογίας, είναι η δυνατότητα επίλυσης του προβλήματος με τη χρήση ενός ιδιαίτερα γνωστού υπολογιστικού εργαλείου. Το εργαλείο αυτό, δεν είναι άλλο από το Solver του Excel, με το οποίο μπορούμε να εφαρμόσουμε και να λύσουμε το πρόβλημα του σχεδιασμού του ωρολόγιου προγράμματος βασισμένοι στην πολυκριτηριακή ανάλυση και το συναινετικό προγραμματισμό. Δεν απαιτείται δηλαδή, κάποιο εξειδικευμένο και δύσκολο στο χειρισμό υπολογιστικό πρόγραμμα, πράγμα που θα εμπόδιζε περαιτέρω την επίλυση.

Είναι λοιπόν, αντιληπτό ότι και στην περίπτωση των ωρολόγιων προγραμμάτων (timetables), ο γραμμικός προγραμματισμός αποτελεί και πάλι μία εγγυημένη και αποτελεσματική επιλογή μεθόδου επίλυσης. Ο συναινετικός προγραμματισμός, συνδυάζοντας τα οφέλη του γραμμικού προγραμματισμού, είναι η κατάλληλη πρόταση για το χειρισμό αυτών των προβλημάτων, καθώς σε συνεργασία με το πολύ λειτουργικό πρόγραμμα-εργαλείου του Excel προσφέρει πολύ ικανοποιητικά αποτελέσματα, που οδηγούν σε ποιοτικά ωρολόγια προγράμματα.

Βιβλιογραφία

1. A. Schaerf. A Survey of Automated Timetabling, *Artificial Intelligence Review* 13(2): 87-127, 1999.
2. E. Burke, Y. Bykov, J. Newall, S. Petrovic. A time-predefined approach to course timetabling, *Yugoslav Journal of Operations Research*, 2003.
3. J. Pearl. *Heuristics: Intelligent search strategies for computer problem solving*. Addison-Wesley, Reading, MA, 1984.
4. Appleby, J. S., Blake, D. V. and Newman, E. A., "Techniques for Producing School Timetables on a Computer and their Application to other Scheduling Problems", *The Computer Journal*, Vol. 3, 237-245, 1960.
5. D. De Werra. An introduction to timetabling. *European Journal of Operational Research*, 19:151{162, 1985.
6. M. R. Garey and D. S. Johnson. *Computers and Intractability /A guide to NP-completeness*. W.H. Freeman and Company, San Francisco, 1979.
7. S. M. Selim. Split vertices in vertex colouring and their application in developing a solution to the faculty timetable problem. *The Computer Journal*, 31(1):76{82, 1988.
8. A. Tripathy. Computerised decision aid for timetabling - A case analysis. *Discrete Applied Mathematics*, 35(3):313{323, 1992.
9. J. A. Breslaw. A linear programming solution to the faculty assignment problem. *Socio-Economic Planning Science*, 10:227{230, 1976.
10. W. Shin and J. A. Sullivan. Dynamic course scheduling for college faculty via zeroone programming. *Decision Science*, 8:711{721, 1977.
11. R. H. McClure and C. E. Wells. A mathematical programming model for faculty course assignment. *Decision Science*, 15:409{420, 1984.
12. A. Tripathy. School timetabling {A case in large binary integer linear programming. *Management Science*, 30(12):1473{1489, 1984.
13. J. A. Ferland and S. Roy. Timetabling problem for university as assignment of activity to resources. *Computers and Operational Research*, 12(2):207{218, 1985.
14. E. Burke, K. Jackson, J. Kingston, R. Weare. *Automated University Timetabling: The state of art*, *The Computer Journal*, 1997.

15. E. Burke, D. Elliman, R. Weare. The Automation of the Timetabling Process in Higher Education, *Journal of Educational Technology Systems*, vol. 23, no. 4, pp. 257-266, Baywood Publishing Company, 1995.
16. B. McCollum, University Timetabling: Bridging the Gap between Research and Practise, *The Proceedings of the 6th International Conference on the Practice and Theory of Automated Timetabling*, pp. 15-35, 2006.
17. S. Petrovic, E. Bruke, University Timetabling, In Leung, J., ed.: *Handbook of Scheduling: Algorithms, Models, and Performance Analysis*. Chapman Hall/CRC Press, pp. 1–23, 2004.
18. A. Wren, Scheduling, Timetabling and Rostering – A special relationship ?, in (Burke and Ross, 1996a), pp. 46-75, 1996.
19. M. Carter and G. Laporte. Recent Developments in Practical Examination Timetabling, in (Burke, and Ross, 1996b), pp. 3-21, 1996a.
20. M. Carter, G. Laporte, and S. Lee. Examination Timetabling: Algorithmic Strategies and Applications, *Journal of the Operational Research Society* vol 47(3), pages 373-383, 1996b.
21. M. Carter and G. Laporte. Recent Developments in Practical Course Timetabling, in (Burke, and Carter, 1998), pp. 3-19, 1998.
22. E. Akkoyunly. A Linear Algorithm for Computing Computer Journal, 16(4), pp. 347-350, 1973.
23. E. Burke, Y. Bykov, and S. Petrovic. A Multicriteria Approach to Examination Timetabling, in *Selected Papers from the 3rd International Conference on the Practice and Theory of Automated Timetabling (PATAT 2000)*, Konstanz, Germany, 16-18 August 2000, Springer-Verlag Lecture Notes in Computer Science, Vol. 2079, E. Burke and W. Erben, Eds., pp. 118-131, 2001a.
24. S. Petrovic, Y. Bykov. A Multiobjective Optimisation Technique for Exam Timetabling Based on Trajectories to appear in (Burke, De Causmaecker, 2003a), 179-192, 2003b.
25. S. Petrovic, and Qu, R. Case-Based Reasoning as a Heuristic Selector in a Hyper-Heuristic for Course Timetabling Problems, *Proceedings of the Sixth International Conference on Knowledge-Based Intelligent Information Engineering Systems and Applied Technologies, KES'02*, Vol. 82., Milan, Italy, September 16-18, pp. 336-340, 2002.
26. E. Burke, B. MacCarthy, S. Petrovic, and Qu, R. Multi-retrieval in a Structured Case Based Reasoning Approach for Course Timetabling problems, Technical Report NOTTCS-TR-2001-7, University of Nottingham, School of Computer Science and IT, 2001d.

27. E. Burke, B. MacCarthy, S. Petrovic, and Qu, R. Knowledge Discovery in a Hyper-Heuristic Using Case-Based Reasoning for Course Timetabling, to appear in (Burke, and De Causmaecker, 2003a), 2003b.
28. E. Burke, S. Petrovic. Recent Research Directions in Automated Timetabling, European Journal of Operational Research - EJOR, 140(2), pp. 266-280, 2002a.
29. E. Burke, S. Petrovic, and Qu, R. Case-Based Heuristic Selection for Examination Timetabling, Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution and Learning - SEAL'02, Singapore, November 18-22, ISBN: 981-04-7523-3, pp 277-281, 2002b.
30. E. Burke, B. MacCarthy, S. Petrovic, and Qu, R. Case-Based Reasoning in Course Timetabling: An Attribute Graph Approach, in Case-Based Reasoning Research and Development, 4th. International Conference on Case-Based Reasoning, ICCBR-2001, Vancouver, Canada, 30July - 2August 2001, Springer-Verlag Lecture Notes in Artificial Intelligence, Vol. 2080, D.W. Aha, I. Watson and Q. Yang, Eds., pp. 90-104, 2001c.
31. V. Bardadym. Computer-Aided School and University Timetabling: The New Wave, Selected and Revised Papers of the 1st International Conference on Practice and Theory of Automated Timetabling, (PATAT 1995), Edinburgh, Springer LNCS 1153, pp. 22-45, 1996.

Παράρτημα

Παράρτημα 1: Activity Template Data Collection Form

Φόρμα συμπλήρωσης για τη συλλογή δεδομένων για τον χρονικό προγραμματισμό των δραστηριοτήτων του Πανεπιστημίου της Μελβούρνης.

Πηγή: <http://www.unimelb.edu.au>

Activity Template Data Collection Form (University of Melbourne)

A potential means for collecting Timetabling information. Shaded data is to be entered into the new student system. Note that one form is required per Activity Type (Lecture, Tutorial etc) so you are likely to have more than one form per subject

Subject Code:	Subject Name:	
Expected Enrolments:	Activity Size (how many students should attend a single stream of this class):	
Activity Type (circle one): Lecture, Tutorial, Workshop, Seminar, Studio, Practical Screening, Concert Class, Instrument Class		
Duration (in minutes):		No of Locations Required:
Room Type Required:		
Campus:	Precinct:	Building:
What Equipment is required for the class?		
Teaching Staff :		
Teaching Week Pattern (cross out those not applicable): Standard/Non-Standard , Sem1/Sem2/Summer/Year Long/Intensive		
Teaching Weeks/Dates:		

Would you like to try to have the same time as last year for this class? If so, what was that day and time?

Do you have a preference to run the class at either of the following times (circle one or more if you do)

Morning / Afternoon / Evening / No preference

Do you have a preference to run the class at a specific time in the week (circle one or more if you do)?

Early week / Mid week / Late Week / Saturday / Sunday

Please add any specific information about how this class is should be timetabled:

Cells shaded green need to be entered in the new student system

Cells shaded yellow need to be entered in the new student system and the Activity Template screens.

Unshaded cells need to be entered in the Activity Template screens.

Στη συνέχεια ακολουθεί η παραπάνω φόρμα συμπληρωμένη από τον διδάσκοντα του μαθήματος Operations Management για το ακαδημαϊκό έτος 2008.

Final Summary	Year of Enrolment: 2008
Subject Code: 325696	Subject Name: Operations Management

SEMESTER 1

EXPECTED ENROLMENTS = 70

L1

Activity Size: 70 - All students will attend the **one** session of this class.

Duration (in minutes): 180

No of Locations Required: 1

Room Type Required: RT:E-LECTURE THEATRE, RT:LECTURE THEATRE

Campus:
CP:MAIN [PARKVILLE]

Precinct:
PR:PROFESSORS WALK

Building:

Equipment/Amenities: AV:DATA PROJECTION, AV:OHP, AV:VCR, IT:COMPUTER, IT:LAPTOP VIDEO SOCKET, IT:PC

Room Features:

Teaching Staff: 027421 - Singh, Prakash Jagat

Teaching Week Pattern: Standard semester week pattern for Semester 1

Morning/Afternoon/Evening Option Selected: A

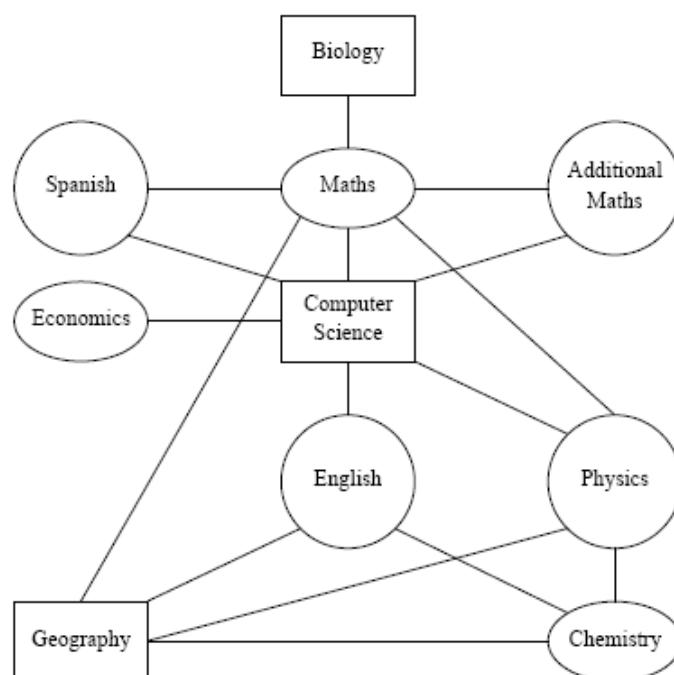
Early/Mid/Late Week or Saturday/Sunday Selected: Late

Any Other Information: Professional coursework subject. Must be scheduled for Thursday 14:15-15:15 to allow attendance and special site visits by part-time students.

Παράρτημα 2: Παράδειγμα επίλυσης με Χρωματισμό Γραφήματος

Στην συνέχεια ακολουθεί ένα υποθετικό παράδειγμα για να γίνει σαφής η χρήση της τεχνικής Χρωματισμού Γραφήματος για την κατάρτιση του ωρολόγιου προγράμματος. Στο συγκεκριμένο υποθετικό παράδειγμα, το κάθε μάθημα αποτελείται από μία μόνο διάλεξη.

Για κάθε μία διάλεξη μαθήματος σχεδιάζεται ένας κόμβος. Και κάθε ζεύγος κόμβων που εκφράζουν διαλέξεις μαθημάτων που δεν επιτρέπεται να προγραμματιστούν την ίδια χρονική περίοδο σχεδιάζουμε μία γραμμή που να ενώνει τους κόμβους του ζεύγους. Στη συνέχεια χρωματίζονται οι κόμβοι, με τρόπο ώστε κανένα ζεύγος κόμβων που ενώνεται με γραμμή να μην εμφανίζεται με το ίδιο χρώμα. Έτσι έχουμε εξασφαλίσει ότι από τη στιγμή που το κάθε χρώμα εκφράζει διαφορετική χρονική περίοδο, δεν θα υπάρχουν επικαλύψεις στις διαλέξεις των μαθημάτων με κοινούς φοιτητές. Στο παράδειγμα αυτό, όπου ο διαφορετικός χρωματισμός δηλώνεται με διαφορετικό σχήμα, μπορούμε να κατανοήσουμε εύκολα ποια μαθήματα πρέπει να προγραμματιστούν σε διαφορετικές χρονικές περιόδους. Δηλαδή είναι εμφανές ότι η Φυσική θα τοποθετηθεί σε διαφορετική χρονική περίοδο σε σχέση με τα Μαθηματικά και τη Γεωγραφία.



Παράρτημα 3: Γενική μεθοδολογία μοντελοποίησης πολυκριτήριων γραμμικών προγραμμάτων

