ΤΕΙ Σερρών
Τμήμα Πληροφορικής & Επικοινωνιών

8/12/2005

Πρόγραμμα Αρχιμήδης ΙΙ

Υποέργο: ΒΕΛΤΙΣΤΗ ΑΥΤΟΜΑΤΗ ΚΑΤΑΡΤΙΣΗ ΩΡΟΛΟΓΙΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ ΜΑΘΗΜΑΤΩΝ ΚΑΙ ΕΞΕΤΑΣΕΩΝ ΓΙΑ ΣΧΟΛΕΣ ΑΕΙ & ΤΕΙ ΜΕ ΧΡΗΣΗ ΜΕΘΟΔΩΝ ΕΞΕΛΙΚΤΙΚΗΣ ΥΠΟΛΟΓΙΣΤΙΚΗΣ

Επιστημονικός Υπεύθυνος : Σπυρίδων Καζαρλής

Αναφορά Προόδου Εργασιών

Κεφάλαιο 1. Ανάπτυξη εφαρμογής για τον αυτόματο υπολογισμό ωρολόγιων προγραμμάτων με βάση τους Γενετικούς Αλγορίθμους (ΓΑ)

1.1 Αντικείμενο και Στόχοι του Έργου

Ο κύριος στόχος αυτού του ερευνητικού έργου είναι η ανάπτυξη ενός ολοκληρωμένου πακέτου λογισμικού (Software) για τον αυτόματο και βέλτιστο υπολογισμό ωρολόγιων προγραμμάτων σπουδών και εξετάσεων για εκπαιδευτικά ιδρύματα με μεγάλο αριθμό σπουδαστών, μαθημάτων, καθηγητών και αιθουσών, όπως τα Α.Ε.Ι. και τα Τ.Ε.Ι. αλλά και τα σχολεία δημοτικής και μέσης εκπαίδευσης της ελληνικής επικράτειας.

Ως γνωστόν η διαδικασία κατασκευής ενός ωρολόγιου προγράμματος σπουδών ή εξετάσεων (Time Table) είναι μία εξαιρετικά πολύπλοκη και επίπονη διαδικασία που η πολυπλοκότητα και η δυσκολία της αυξάνεται εκθετικά με τον αριθμό των παραμέτρων του προβλήματος που είναι ο αριθμός των σπουδαστών, των μαθημάτων, των καθηγητών και των αιθουσών.

Η δυσκολία της επίλυσης τέτοιων προβλημάτων έγκειται σε δύο βασικούς λόγους : α) Στον μεγάλο αριθμό των συνδυασμών / λύσεων που πρέπει να ελεγχθούν ώστε να βρεθεί μία ικανοποιητική λύση, ή πιο αισιόδοξα, μία κατά το δυνατό βέλτιστη λύση με βάση τους στόχους που τίθενται, και β) στον μεγάλο αριθμό των περιορισμών που πρέπει να ικανοποιούνται από μία υποψήφια λύση, που μπορούν να αφορούν την διαθεσιμότητα και τις ώρες των καθηγητών, την διαθεσιμότητα και την καταλληλότητα των αιθουσών, την μη επικάλυψη ωρών σε μαθήματα που παρακολουθούνται από σπουδαστές του ιδίου εξαμήνου, την ισοκατανομή των ωρών στους καθηγητές και στις ημέρες της εβδομάδας, κ.λ.π.

Συνδυαστικά προβλήματα (combinatorial problems) αυτού του τύπου θεωρούνται στην διεθνή ακαδημαϊκή κοινότητα ως NP complete, δηλαδή προβλήματα που δεν επιλύονται με γνωστές μεθόδους σε πολυωνυμικό χρόνο (δηλαδή σε χρόνο που να είναι πολυωνυμική συνάρτηση του αριθμού των παραμέτρων) απαιτώντας εκθετικούς χρόνους για την επίλυσή τους, γεγονός που καθιστά αδύνατη την επίλυση τέτοιων προβλημάτων όταν ο αριθμός των παραμέτρων είναι μεγάλος.

Η επίλυση του προβλήματος κατασκευής ενός ωρολόγιου προγράμματος συνήθως ανατίθεται σε κάποιον εκ των καθηγητών ή σε ομάδα καθηγητών, οι οποίοι συνήθως βασίζονται στο ωρολόγιο πρόγραμμα της προηγούμενης ακαδημαϊκής περιόδου, πάνω στο οποίο πραγματοποιούν μικρές μεταβολές, ανάλογα με τις εκάστοτε ανάγκες και συνήθως

αναλώνουν πολύ χρόνο (αρκετές ημέρες) για την εξέλιξη ενός και μόνον προγράμματος. Βέβαια ένα τέτοιο πρόγραμμα πολλές φορές εξελίσσεται τόσο ώστε να ικανοποιήσει τους βασικούς περιορισμούς του προβλήματος, όπως η επικάλυψη ωρών για αίθουσες και καθηγητές, δημιουργώντας πολλές φορές προβλήματα και παράπονα από την μεριά τόσο των καθηγητών όσο και των σπουδαστών λόγω της μη επίτευξης άλλων στόχων όπως είναι η ελαχιστοποίηση των κενών ωρών ή η ελαχιστοποίηση των μετακινήσεων καθηγητών και σπουδαστών μέσα στον χώρο του ιδρύματος (campus). Επιπλέον, ένα πρόγραμμα που κατασκευάζεται κατ΄ αυτό τον τρόπο, δεν μπορεί να ελεγχθεί για το κατά πόσο είναι βέλτιστο, λόγω του μεγάλου αριθμού συνδυαστικών λύσεων που υπάρχουν και που δεν μπορούν να κατασκευαστούν και να ελεγχθούν από ανθρώπινο δυναμικό σε εύλογο χρόνο.

Για την υλοποίηση λοιπόν ενός λογισμικού που θα είναι σε θέση να παράγει αυτόματα βέλτιστες λύσεις ωρολόγιων προγραμμάτων, απαιτείται η χρήση σύγχρονων επιστημονικών εργαλείων αναζήτησης λύσεων και βελτιστοποίησης σε συνδυαστικά προβλήματα. Ένα τέτοιο σύγχρονο εργαλείο είναι και οι Γενετικοί Αλγόριθμοι (Genetic Algorithms) που ειδικεύονται στην επίλυση δύσκολων συνδυαστικών προβλημάτων.

1.2 Χρήση Σύγχρονων Εργαλείων Βελτιστοποίησης

Οι Γενετικοί Αλγόριθμοι (Genetic Algorithms ή GAs) αποτελούν αλγόριθμους γενικής βελτιστοποίησης (Global Optimization Algorithms) και αναζήτησης (Search Algorithms) γενικής εφαρμογής. Η ανακάλυψη των Γενετικών Αλγορίθμων εμπίπτει χρονικά στο ρεύμα έρευνας των τελευταίων δεκαετιών που ήταν σαφώς προσανατολισμένο στην υλοποίηση συστημάτων και τεχνικών με σκοπό την προσομοίωση συγκεκριμένων βιολογικών συστημάτων και διαδικασιών καθώς και τον τρόπο σκέψης των ανθρώπων. Σε αυτό το ρεύμα εμπίπτουν και τα Τεχνητά Νευρωνικά Δίκτυα (Artificial Neural Networks), η Τεχνητή Ευφυΐα (Artificial Intelligence) και τα Συστήματα Ασαφούς Λογικής (Fuzzy Logic Systems) που αποτελούν τους κύριους εκπροσώπους ενός νέου επιστημονικού τομέα που ονομάζεται Υπολογιστική Ευφυΐα (Computational Intelligence). Οι Γενετικοί Αλγόριθμοι βασίζονται σε αρχές εμπνευσμένες από τους πραγματικούς μηχανισμούς της Γενετικής και της εξέλιξης των ειδών (θεωρία του Δαρβίνου) που διέπουν τα φυσικά βιολογικά συστήματα.

Όπως είναι γνωστό στα φυσικά βιολογικά συστήματα ισχύει οι νόμος της φυσικής επιλογής, μέσα από το μηχανισμό της επιβίωσης των καλύτερων (survival of the fittest), για την επιλογή ατόμων προς αναπαραγωγή. Η αναπαραγωγή σε τέτοια συστήματα γίνεται συνήθως με την ανάμιξη των γενετικών χαρακτηριστικών δύο γονέων, που επιλέγονται με βάση τον παραπάνω νόμο, και οδηγεί στη γένεση νέων ατόμων στον πληθυσμό. Ο ανασυνδυασμός των γενετικών χαρακτηριστικών των γονέων μαζί με την μικρή αλλά όχι μηδενική πιθανότητα μετάλλαξης των χαρακτηριστικών αυτών από εξωτερικούς παράγοντες του περιβάλλοντος, είναι που δίνουν την πιθανότητα δημιουργίας νέων βελτιωμένων χαρακτηριστικών αλλά και την πιθανότητα δημιουργίας χειρότερων χαρακτηριστικών στον πληθυσμό των απογόνων. Σε συνδυασμό όμως με τον νόμο επιβίωσης των καλύτερων η εξέλιξη που πραγματοποιείται έχει κατεύθυνση προς τη σταδιακή βελτίωση των χαρακτηριστικών του πληθυσμού και την ολοένα και καλύτερη προσαρμογή τους στο περιβάλλον. Αυτή η δυναμική προσαρμογής και βελτίωσης καθιστά τους πληθυσμούς βιώσιμους στις μεταβολές του περιβάλλοντος και συμβάλλει στην βελτίωση των χαρακτηριστικών των ειδών.

Αυτή τη δυναμική των πληθυσμών προσπαθούν να εξομοιώσουν και οι Γενετικοί Αλγόριθμοι με απώτερο σκοπό όχι τόσο τη μοντελοποίηση και μελέτη των φυσικών βιολογικών συστημάτων (σκοπό με τον οποίο ξεκίνησαν) αλλά την αναζήτηση και εξέλιξη λύσεων σε προβλήματα βελτιστοποίησης και εκμάθησης. Έτσι ο βαθμός προσαρμογής στο

περιβάλλον αντιστοιχίζεται σε τιμές μιας προς βελτιστοποίηση **Συνάρτησης Ποιότητας** (Fitness Function), ο πληθυσμός ατόμων εξομοιώνεται από ένα πληθυσμό λύσεων του προβλήματος, τα χαρακτηριστικά των ατόμων αντιστοιχίζονται σε χαρακτηριστικά των λύσεων και ο ανασυνδυασμός των χαρακτηριστικών αυτών καθώς και ο τρόπος επιλογής γονέων από τον πληθυσμό γίνονται τεχνητά, προσομοιώνοντας όμως τις αντίστοιχες διαδικασίες της φύσης κατά το δυνατόν. Έτσι δημιουργείται ένα τεχνητό βιολογικό σύστημα που έχει τη δυνατότητα εξέλιξης βάση των νόμων της επικράτησης του καλύτερου και της στοχαστικής ανασυνδυαστικής αναπαραγωγής μεταξύ ατόμων ενός πληθυσμού που έχει τη δυνατότητα να αυτο-οδηγείται στην παραγωγή ενός βέλτιστου ατόμου που αποτελεί και τη βέλτιστη λύση στο κάθε φορά κωδικοποιημένο πρόβλημα.

Και παρόμοια με τη φύση, όπου οι ίδιοι νόμοι ισχύουν για όλα τα βιολογικά συστήματα έτσι και οι τεχνητοί Γ.Α. μπορούν να εφαρμοστούν σε όλα τα προβλήματα βελτιστοποίησης θέτοντας ελάχιστες προαπαιτήσεις και με ελάχιστες προσαρμογές.

Το βασικό πλεονέκτημα των Γ.Α είναι ότι δεν θέτουν καμία προαπαίτηση για την φύση του προβλήματος (π.χ. ύπαρξη συνέχειας ή παραγώγων), παρά μόνο την ύπαρξη μίας αντικειμενικής συνάρτησης που θα αποτιμά κάθε παραγόμενη λύση. Επίσης δεν απαιτούν την ύπαρξη καμίας εκ των προτέρων γνώσης για τον τρόπο επίλυσης του κάθε προβλήματος. Η μεγάλη τους αποτελεσματικότητα, δε, οφείλεται στο βέλτιστο συνδυασμό της εξερεύνησης του χώρου λύσεων (exploration) και της εκμετάλλευσης των λύσεων που βρέθηκαν, για την παραγωγή ακόμα καλύτερων λύσεων (exploitation). Αυτά συντελούνται μέσω των γενετικών τελεστών που εφαρμόζονται στον εξελισσόμενο πληθυσμό ο οποίος εισάγει με τη σειρά του, λόγω του αριθμού των λύσεων που περιέχει και που εξελίσσονται παράλληλα, μία ενδογενή παραλληλοποίηση της διαδικασίας της αναζήτησης λύσεων (implicit parallelism). Επίσης οι Γ.Α. είναι σε θέση να παρέχουν **όχι μόνον μία βέλτιστη λύση αλλά έναν ολόκληρο πληθυσμό (π.χ. 100) εναλλακτικών υπο-βέλτιστων λύσεων** τις οποίες θα μπορεί ο χρήστης του λογισμικού να δει και από αυτές να επιλέξει την πλέον κατάλληλη.

Οι Γενετικοί Αλγόριθμοι έχουν ήδη εφαρμοστεί με επιτυχία σε ένα μεγάλο σύνολο πραγματικών προβλημάτων που συναντώνται στην επιστήμη, την μηχανική, την βιομηχανία αλλά και στην καθημερινή ζωή. Ενδεικτικά μπορούμε να αναφέρουμε κάποιες από τις εφαρμογές τους :

- Βελτιστοποίηση Μαθηματικών Συναρτήσεων με ή χωρίς περιορισμούς
- Επίλυση Προβλημάτων των Συστημάτων Ισχύος (Economic Dispatch, Unit Commitment)
- Επίλυση Προβλημάτων Κοπής και Συσκευασίας (Cutting and Packing)
- Επίλυση προβλημάτων σχεδιασμού γραμμής παραγωγής σε βιομηχανίες (Job Shop Scheduling)
- Επίλυση προβλημάτων σχεδιασμού και δρομολόγησης (routing and scheduling)
- Εφαρμογή ως αλγόριθμος εκπαίδευσης σε Τεχνητά Νευρωνικά Δίκτυα
- Επίλυση προβλημάτων αναγνώρισης προτύπων (Pattern recognition)
- Επίλυση προβλημάτων βέλτιστης τοπολογίας δικτύων (επικοινωνίας, συγκοινωνίας, ύδρευσης, διανομής αερίου κ.λ.π.)
- Επίλυση προβλημάτων στην σχεδίαση κυκλωμάτων πολύ υψηλής κλίμακας ολοκλήρωσης (VLSI)
- Επίλυση προβλημάτων κατανομής αγαθών (Resource scheduling) π.χ. κατανομή χρόνου CPU σε έναν Η/Υ με πολυδιεργασία (multi-tasking)
- Εφαρμογή στην βιομηχανική σχεδίαση (π.χ. σχεδίαση αεροδυναμικού σχήματος αεροσκαφών) και στην σχεδίαση ελεγκτών (Controllers).
- Εφαρμογή στην σχεδίαση και βελτιστοποίηση Συστημάτων Ασαφούς Λογικής (Fuzzy Logic Systems)

- Εφαρμογή στην Ρομποτική (Εύρεση βέλτιστης τροχιάς βραχίονα, εύρεση βέλτιστων μονοπατιών κίνησης κ.λ.π.)
- Εφαρμογή στην εύρεση βέλτιστων τοπολογιών σε Εξελισσόμενο Υλικό (Evolvable Hardware) για την επίτευξη επιθυμητής συμπεριφοράς.

1.3 Βελτιστοποίηση Ωρολόγιων Προγραμμάτων με Γενετικούς Αλγόριθμους

Η εφαρμογή Γενετικών Αλγορίθμων σε προβλήματα Ωρολόγιου Προγράμματος (Time Tabling Problems) είναι ένα αντικείμενο ακαδημαϊκής έρευνας και στην Ελλάδα (πρόσφατα) αλλά κυρίως στην διεθνή ακαδημαϊκή κοινότητα από το 1990 [CoDM90]. Έκτοτε αποτελεί ένα πεδίο έρευνας στα πλαίσια της επιστήμης της Εξελικτικής Υπολογιστικής (Evolutionary Computation) με ραγδαία ανάπτυξη. Επίσης το πεδίο έρευνας για αυτοματοποιημένα ωρολόγια προγράμματα έχει και το δικό του διεθνές συνέδριο από το 1995 (PATAT). Στο δεύτερο διεθνές συνέδριο για αυτοματοποιημένα ωρολόγια προγράμματα (PATAT '97) σχεδόν το ένα τέταρτο των εργασιών χρησιμοποιούσαν κάποια μέθοδο εξελικτικής υπολογιστικής όπως οι Γ.Α..
Τα προβλήματα ωρολόγιου προγράμματος μπορούν να χωριστούν σε 4 κατηγορίες που είναι :

1. Ωρολόγια Προγράμματα Σχολείων (School Time Tabling)
2. Ωρολόγια Προγράμματα Διαλέξεων (Lecture Time Tabling)
3. Ωρολόγια Προγράμματα Εξετάσεων (Exam Time Tabling)
4. Ωρολόγια Προγράμματα Υπαλλήλων (Employee Time Tabling)

Η κατανομή των εργασιών σχετικών με την χρήση Εξελικτικών Μεθόδων για την επίλυση προβλημάτων των τεσσάρων κατηγοριών στα τελευταία διεθνή συνέδρια φαίνεται στον παρακάτω πίνακα :

Κατηγορία	***Αριθμός Εργασιών***							
	PPSN 1992	ICGA 1993	PPSN 1994	PATAT 1995	ICGA 1995	PPSN 1996	ICGA 1997	PATAT 1997
School Timetabling	1							1
Lecture Timetabling			1	4	1			5
Exam Timetabling				3		1		3
Employee Timetabling		1						2

Στις δημοσιευμένες εργασίες υπάρχει μεγάλη συγκέντρωση γύρω από την επίλυση προβλημάτων ωρολογίου προγράμματος των Πανεπιστημίων (University Time Tabling) κυρίως λόγω του ότι η έρευνα διεξάγεται κατά το μεγαλύτερο ποσοστό σε αυτά, και επομένως οι ερευνητές μπορούν εύκολα να συγκεντρώσουν πραγματικά δεδομένα από τα ωρολόγια προγράμματα των Πανεπιστημίων αλλά και να ελέγξουν την αποτελεσματικότητα των αλγορίθμων που αναπτύσσουν.
Στις περισσότερες περιπτώσεις προβλημάτων ωρολόγιου προγράμματος που αντιμετωπίζονται στην βιβλιογραφία, η κλίμακες των προβλημάτων είναι μεγάλες. Για παράδειγμα στην εργασία [MaRe97] χρησιμοποιείται ένα πρόβλημα με 7000 σπουδαστές, 169 διαλέξεις, 211 μαθήματα, 98 καθηγητές και 396 διαθέσιμες ώρες.

Μία ανασκόπηση όλης της σχετικής βιβλιογραφίας δημοσιεύτηκε από τον M.W.Carter το 1986 στην εργασία [Cart86], και επαναδημοσιεύτηκε αναθεωρημένη το 1995 [CaLa95] και το 1997 [CaLa97].
Ορισμένες από τις εργασίες έχουν οδηγήσει στην κατασκευή ολοκληρωμένων συστημάτων επίλυσης ωρολόγιων προγραμμάτων, οι οποίες πλέον αποτελούν έτοιμες εμπορικές εφαρμογές. Τέτοιες περιπτώσεις είναι :

- Το σύστημα **ACT** (Automated Class Timetabler) που αποτελεί ένα σύστημα επίλυσης ωρολόγιων προγραμμάτων Πανεπιστημίων τελευταίας τεχνολογίας, που τρέχει σε περιβάλλον Windows και αναπτύχθηκε στην Κορέα. Έχει εφαρμοστεί με επιτυχία σε 15 Πανεπιστήμια στην Κορέα και το μεγαλύτερο πρόβλημα στο οποίο εφαρμόστηκε είχε 70 τμήματα, 700 καθηγητές και 4000 μαθήματα. Χρησιμοποιώντας ένα Pentium στα 133 MHz μπορεί να κατασκευάσει ωρολόγια προγράμματα σε περίπου 20 λεπτά και χρησιμοποιεί οθόνη γραφικών για την εισαγωγή δεδομένων και την εμφάνιση των αποτελεσμάτων.
- Το σύστημα **Neeps and Tatties** του Πανεπιστημίου Napier στο Εδιμβούργο (Napier University, in Edinburgh) [PaLCP94], [PaRC97]. Αν και βασίζεται σε διαφορετικές αρχές απ' ότι το προηγούμενο σύστημα, εφαρμόστηκε με επιτυχία στο Πανεπιστήμιο Napier παράγοντας ωρολόγια προγράμματα από το 1997 για 2067 μαθήματα, σε 45 ώρες και 183 αίθουσες, με 69 καθηγητές και 978 σπουδαστές. Το σύστημα αυτό βελτιστοποιεί περίπου 12 διαφορετικά κριτήρια / περιορισμούς και είναι σε θέση να παράγει ένα ωρολόγιο πρόγραμμα που να ικανοποιεί τους περιορισμούς μέσα σε μία ώρα. Στη συνέχεια εκτελείται απ' άπειρον προσπαθώντας να βελτιστοποιήσει τα υπόλοιπα κριτήρια. Στην πράξη το σύστημα αφέθηκε να λειτουργεί επί τρεις ημέρες (το 1997). Το πλεονέκτημά του είναι ότι μπορεί να τροφοδοτηθεί με παλιά προγράμματα και ότι ο χρήστης μπορεί δυναμικά να μεταβάλλει τα βάρη (σημαντικότητα) των περιορισμών και των κριτηρίων.
- Ένα άλλο σύστημα που αναπτύχθηκε στο Πανεπιστήμιο του Εδιμβούργου είναι το **GATT** (Genetic Algorithm Time Tabler), για τον προγραμματισμό των εξετάσεων. Το λογισμικό αυτό χρησιμοποιήθηκε επίσης από το Harvard Business School για τις εξετάσεις, και δοκιμάστηκε και σε ένα σχολείο του Βελγίου σε ένα πρόβλημα 26 καθηγητών και 318 μαθημάτων.

Πρόσφατα υπήρξαν και δημοσιεύσεις ελλήνων ερευνητών πάνω στην εφαρμογή Γ.Α. σε προβλήματα ωρολογίου προγράμματος [AdAr99] όπου επιλύονται προβλήματα έως και 180 μαθημάτων (events).

1.4 Υλοποίηση

Στο έργο λοιπόν αυτό συμπεριλαμβάνεται η ανάπτυξη ενός Γενετικού Αλγορίθμου που θα αναλάβει την αναζήτηση βέλτιστων λύσεων σε προβλήματα ωρολόγιου προγράμματος τα οποία θα καθορίζονται δυναμικά από τον χρήστη. Η ανάπτυξη αυτή προϋποθέτει την μελέτη της διεθνούς επιστημονικής βιβλιογραφίας πάνω σε αυτό το αντικείμενο, καθώς και μία ερευνητική δουλειά για την κατασκευή, τον έλεγχο και την τελική ρύθμιση του Γ.Α. ώστε να εργάζεται αποτελεσματικά και να παρέχει βέλτιστες λύσεις σε πολλαπλές περιπτώσεις προβλημάτων ωρολόγιου προγράμματος που θα τεθούν ως προβλήματα ελέγχου (benchmarks) και θα προέρχονται από πραγματικές περιπτώσεις ωρολόγιων προγραμμάτων συγκεκριμένων ιδρυμάτων.

Η εφαρμογή θα διαθέτει επίσης και ένα τμήμα επικοινωνίας με τον χρήστη (interface) μέσω του οποίου ο χρήστης θα καθορίζει δυναμικά τις παραμέτρους του ωρολόγιου προγράμματος (π.χ. αριθμός καθηγητών, σπουδαστών και αιθουσών, συσχετισμοί καθηγητών

με μαθήματα, εξοπλισμός αιθουσών, ώρες ανά μάθημα την εβδομάδα, μέγιστος αριθμός ωρών ανά καθηγητή, χωρητικότητα αιθουσών κ.λ.π.). Μέσω αυτού του τμήματος θα γίνεται και η εμφάνιση της βέλτιστης λύσης στον χρήστη, ή ακόμα και εναλλακτικών υπο-βέλτιστων λύσεων, ώστε να επιλεχθεί η επιθυμητή. Επίσης ο χρήστης θα μπορεί να προβαίνει σε μεταβολές του προγράμματος και να ζητά επαν-επίλυση του προβλήματος ή έλεγχο ικανοποίησης των περιορισμών.

1.4.1 Βασικά ζητήματα υλοποίησης

Για την υλοποίηση ενός ΓΑ, εκτός από την ανάπτυξη του βασικού αλγορίθμου, θα πρέπει ειδικότερα να αποφασιστεί η τεχνική που θα ακολουθηθεί και εν συνεχεία να γίνει η κωδικοποίηση σε πρόγραμμα Η/Υ των εξής σημείων :

Α. Μέθοδος Αναπαράστασης των λύσεων στον ΓΑ (κωδικοποίηση των λύσεων). Δηλαδή πρέπει να αποφασιστεί με ποιον ακριβώς τρόπο θα αναπαρίσταται μία λύση ενός προβλήματος TimeTabling ως συμβολοσειρά γονότυπου, ώστε να μπορεί να την χειριστεί ο ΓΑ και να εφαρμόσει σε αυτήν τους γενετικούς τελεστές (π.χ. Crossover και Mutation).

Β. Η κατασκευή της Συνάρτησης Ποιότητας (Fitness Function) η οποία αξιολογεί κάθε παραγόμενη λύση του ΓΑ. Δηλαδή αποκωδικοποιεί την προς αξιολόγηση λύση που παρήγαγε ο ΓΑ, σε μορφή κανονικού ωρολόγιου προγράμματος, και εν συνεχεία υπολογίζει την Αντικειμενική Ποιότητα της λύσης (Objective Fitness Value) που αφορά τα κριτήρια βελτιστοποίησης και ακολούθως ελέγχει την λύση για παραβιάσεις των περιορισμών, υπολογίζοντας τιμές ποινών (Penalty values) για κάθε παραβίαση. Το άθροισμα Objective + Penalty Value μας δίνει την συνολική ποιότητα της λύσης

1.4.2 Κωδικοποίηση των λύσεων – Μέθοδος Αναπαράστασης Λύσεων

Ο μηχανισμός του ΓΑ παράγει συνέχεια λύσεις του προβλήματος με μεθόδους προσομοιωμένης γενετικής εξέλιξης. Για να λειτουργήσει ο μηχανισμός αναπαραγωγής λύσεων του ΓΑ καθώς και οι Γενετικοί Τελεστές, θα πρέπει οι λύσεις που παράγει και χειρίζεται ο ΓΑ να είναι κωδικοποιημένες ως συμβολοσειρές. Η κωδικοποίηση γίνεται χρησιμοποιώντας ένα σετ συμβόλων που συνήθως είναι το σετ των δυαδικών ψηφίων (0,1). Επειδή δεν υπάρχει ενιαίος τρόπος κωδικοποίησης όλων των προβλημάτων με δυαδικές συμβολοσειρές, θα πρέπει ο εφαρμοστής των ΓΑ να αποφασίζει κάθε φορά για τον τρόπο αναπαράστασης των λύσεων. Ο τρόπος αναπαράστασης των λύσεων είναι πολύ σημαντικός για την απόδοση του ΓΑ και θα πρέπει να ικανοποιεί μερικά κριτήρια :

Α. Η κωδικοποίηση πρέπει να είναι αμφι-μονοσήμαντη δηλαδή κάθε λύση να έχει μία μόνο δυαδική αναπαράσταση, και κάθε αναπαράσταση να αποκωδικοποιείται σε μία μοναδική λύση.

Β. Θα πρέπει να παράγει συμβολοσειρές με όσο το δυνατόν μικρότερο μήκος, που σημαίνει ότι δεν πρέπει να συμμετέχουν στον γονότυπο άχρηστα bits που δεν φέρουν πληροφορία.

Γ. Θα πρέπει μεμονωμένα χαρακτηριστικά μίας λύσης να αντιπροσωπεύονται από κατά το δυνατόν συνεχόμενα bits στην αναπαράσταση, έτσι ώστε να μορφοποιούν τα λεγόμενα building blocks ή schemas (θεωρία των ΓΑ) τα οποία εξελίσσονται παράλληλα.

Δ. Θα πρέπει η αναπαράσταση να υποβοηθά την λειτουργία των γενετικών τελεστών, και να είναι συμβατή με αυτούς. Αυτό σημαίνει ότι η εφαρμογή των γενετικών τελεστών (crossover & mutation) πάνω σε έγκυρες κωδικοποιημένες λύσεις θα πρέπει επίσης να δίνει έγκυρη λύση.

Με βάση τις παραπάνω αρχές αλλά και μεθόδους που έχουν δημοσιευτεί στην βιβλιογραφία προχωρήσαμε στην εξής μέθοδο κωδικοποίησης / αναπαράστασης λύσεων :

Στην ουσία θα πρέπει μία λύση Ωρολόγιου Προγράμματος να περιλαμβάνει την πλήρη περιγραφή της χρονικής τοποθέτησης των προκαθορισμένων συμβάντων (events / μαθήματα) μέσα στον χρονικό ορίζοντα προγραμματισμού (εβδομάδα), καθώς και την διασύνδεση του συμβάντος με τα απαραίτητα «αγαθά» (resources) δηλαδή με συγκεκριμένους καθηγητές και αίθουσα.
Κωδικοποιώντας λοιπόν με X bits την χρονική τοποθέτηση, με T bits τον καθηγητή και με R bits την αίθουσα και πολλαπλασιάζοντας τα E=X+T+R bits επί τον αριθμό των μαθημάτων N παίρνουμε την συνολική συμβολοσειρά μιας λύσης με NxE bits. Αυτή θα μπορούσε να θεωρηθεί ως μία κανονική / αναλυτική / φυσιολογική κωδικοποίηση.
Παρουσιάζει όμως ένα πρόβλημα: αν αφήσουμε τον ΓΑ να εξελίξει γενετικά την χρονική τοποθέτηση των μαθημάτων μέσα στον χρονικό ορίζοντα του προγράμματος, είναι σίγουρο ότι οι τοποθετήσεις θα παρουσιάζουν χρονικές επικαλύψεις και άρα θα συνιστούν παραβιάσεις των περιορισμών του προβλήματος. Θα πρέπει λοιπόν ο ΓΑ να εντοπίσει μία τέτοια διευθέτηση των χρονικών τοποθετήσεων των μαθημάτων όπου δεν θα υπάρχει καμία χρονική αλληλοεπικάλυψη. Το πρόβλημα έτσι καθίσταται εξαιρετικά δύσκολο για τον ΓΑ ειδικά μάλιστα εάν σκεφτούμε και την ύπαρξη και των άλλων περιορισμών (Καθηγητών, Αιθουσών κ.λ.π.).
Για τους παραπάνω λόγους υιοθετήθηκε μία μέθοδος αναπαράστασης λύσεων που λειτουργεί όχι σε αναλυτικό αλλά σε αφαιρετικό επίπεδο η οποία δεν περιγράφει αναλυτικά την χρονική τοποθέτηση των μαθημάτων αλλά αφαιρετικά.
Συγκεκριμένα ο ΓΑ καθορίζει στις παραγόμενες κωδικοποιημένες λύσεις μόνο την ημέρα της εβδομάδας στην οποία θα διεξαχθεί το κάθε μάθημα, αλλά όχι την ώρα της ημέρας. Η τελική ανάθεση της ώρας διεξαγωγής του κάθε μαθήματος γίνεται από ένα δεύτερο αλγόριθμο που λειτουργεί με ευθύ, ντετερμινιστικό τρόπο. Ο αλγόριθμος αυτός τοποθετεί με την σειρά τα μαθήματα της κάθε μέρας σε συγκεκριμένες ώρες, φροντίζοντας ώστε να μην επικαλύπτονται χρονικά μεταξύ τους, και ταυτόχρονα φροντίζει για την ικανοποίηση ενός αριθμού περιορισμών, όπως η διαθεσιμότητα του Καθηγητή και της Αίθουσας.
Με αυτόν τον τρόπο ένας σημαντικός αριθμός περιορισμών ικανοποιούνται απευθείας από την μέθοδο αναπαράστασης λύσεων.
Επειδή βέβαια είναι εξαιρετικά σημαντική η σειρά με την οποία θα επιχειρηθεί να τοποθετηθούν τα μαθήματα της ημέρας σε χρονοθυρίδες, προβλέψαμε την ύπαρξη και μίας ακόμα παραμέτρου η οποία θα κωδικοποιείται μέσα στη λύση για κάθε μάθημα (και θα εξελίσσεται γενετικά), την Προτεραιότητα Διεκπεραίωσης.
Έτσι ο ντετερμινιστικός αλγόριθμος ανάθεσης χρονοθυρίδων στα μαθήματα ανά ημέρα, διεκπεραιώνει τα μαθήματα σύμφωνα με την Προτεραιότητα του κάθε μαθήματος. Μαθήματα με υψηλή προτεραιότητα (μικρό νούμερο) επιχειρείται να τοποθετηθούν πρώτα στο χρονοδιάγραμμα της κάθε ημέρας, ενώ μαθήματα με χαμηλή προτεραιότητα (μεγάλο νούμερο) διεκπεραιώνονται τελευταία και τοποθετούνται συνήθως προς το τέλος της ημέρας.
Έτσι λοιπόν η Αφαιρετική Κωδικοποίηση για κάθε μάθημα χρησιμοποιεί :

R bits	για την αίθουσα
TxM bits	για τους καθηγητές, όπου T τα bits για κωδικοποίηση ενός καθηγητή και M ο μέγιστος αριθμός καθηγητών ανά μάθημα.
D bits	για την ημέρα του προγράμματος και
P bits	για την Προτεραιότητα Διεκπεραίωσης του κάθε μαθήματος.

και πολλαπλασιάζοντας τα E=R+TxM+D+P bits επί τον αριθμό των μαθημάτων N παίρνουμε την συνολική συμβολοσειρά μιας λύσης με NxE bits.

1.4.3 Η Συνάρτηση Ποιότητας - Περιορισμοί και Στόχοι

Κάθε παραγόμενη λύση θα πρέπει να αξιολογείται από τον ΓΑ βάση μίας Συνάρτησης Ποιότητας (Fitness Function), η οποία παίρνει την παραχθείσα λύση, την αποκωδικοποιεί και την αξιολογεί, τόσο ως προς τον βαθμό ικανοποίησης των στόχων (objectives) όσο και ως προς την ικανοποίηση των περιορισμών.

Οι περιορισμοί που ελέγχονται στην Συνάρτηση Ποιότητας είναι οι εξής :

- **Μέγιστος αριθμός ωρών/περιόδων ανά ημέρα** : Το ημερήσιο πρόγραμμα δεν μπορεί να ξεπερνάει ένα συγκεκριμένο αριθμό ωρών ανά ημέρα.
- **Κάθε Τμήμα (Class) μπορεί να διεξαχθεί σε συγκεκριμένες αίθουσες** : Για κάθε Τμήμα καθορίζεται μία λίστα με τις αίθουσες στις οποίες μπορεί να διεξαχθεί.
- **Ώρες Διαθεσιμότητας Αιθουσών** : Για κάθε αίθουσα καθορίζεται ποιες ώρες για κάθε ημέρα της περιόδου (εβδομάδας) είναι διαθέσιμη. Τις υπόλοιπες ώρες η αίθουσα θεωρείται μη χρησιμοποιήσιμη (κλειστή).
- **Ανάληψη Τμημάτων από Καθηγητές** : Για κάθε Καθηγητή καθορίζεται μία λίστα με τα Τμήματα που μπορεί να αναλάβει.
- **Μέγιστος Αριθμός Ωρών Διδασκαλίας ανά Καθηγητή** : Για κάθε Καθηγητή καθορίζεται ο μέγιστος αριθμός ωρών που μπορεί να αναλάβει μέσα στην περίοδο (εβδομάδα).
- **Ώρες Διαθεσιμότητας Καθηγητών** : Για κάθε Καθηγητή καθορίζεται ποιες ώρες για κάθε ημέρα της περιόδου (εβδομάδας) είναι διαθέσιμος. Τις υπόλοιπες ώρες ο Καθηγητής θεωρείται μη διαθέσιμος (π.χ. απασχολείται αλλού).
- **Επικάλυψη Αιθουσών** : Η ίδια αίθουσα δεν είναι δυνατόν να καταλαμβάνεται από δύο τμήματα ταυτόχρονα.
- **Επικάλυψη Καθηγητών** : Ο ίδιος Καθηγητής δεν μπορεί να διδάσκει σε δύο τμήματα ταυτόχρονα
- **Επικάλυψη Τμημάτων του ίδιου Εξαμήνου** : Δύο τμήματα του ίδιου εξαμήνου δεν μπορεί να γίνονται ταυτόχρονα.
- **Χρονική Απόσταση πολλαπλών ωρών του ίδιου Τμήματος** : αν ένα Τμήμα έχει πολλαπλές ώρες μέσα στην περίοδο (εβδομάδα) π.χ. 2 δίωρα , θα πρέπει αυτές οι ώρες να ανατεθούν σε διαφορετικές ημέρες με κάποια απόσταση ημερών μεταξύ τους. Η ελάχιστη αυτή απόσταση (σε ημέρες) καθορίζεται για κάθε Τμήμα.
- **Ελάχιστος Χρόνος Μεταφοράς Τμημάτων του ίδιου Εξαμήνου** : ο χρόνος μεταφοράς των Τμημάτων από μία αίθουσα σε μία άλλη (Travel Time) μεταξύ δύο διαδοχικών μαθημάτων /τμημάτων του ίδιου εξαμήνου δεν πρέπει να ξεπερνά ένα όριο (π.χ. ένα τέταρτο της ώρας) που καθορίζεται για όλο το πρόγραμμα.
- **Ελάχιστος Χρόνος Μεταφοράς Καθηγητών** : ο χρόνος μεταφοράς των Καθηγητών από μία αίθουσα σε μία άλλη (Travel Time) μεταξύ δύο διαδοχικών μαθημάτων / τμημάτων δεν πρέπει να ξεπερνά ένα όριο (π.χ. ένα τέταρτο της ώρας) που καθορίζεται για όλο το πρόγραμμα.

Οι στόχοι τους οποίους ο Γ.Α. θα βελτιστοποιεί, ελέγχοντας τον βαθμό ικανοποίησής τους μέσω της Συνάρτησης Ποιότητας θα είναι οι εξής :

- **Ελαχιστοποίηση Κενών Ωρών Καθηγητών** : στόχος είναι να ελαχιστοποιηθούν οι κενές ώρες των καθηγητών μέσα σε όλο το πρόγραμμα.
- **Ελαχιστοποίηση Κενών Ωρών Αιθουσών** : στόχος είναι να ελαχιστοποιηθούν οι ώρες που οι αίθουσες παραμένουν κενές (αχρησιμοποίητες) μέσα σε όλο το πρόγραμμα. Ο

στόχος αυτός ταυτίζεται με την ελαχιστοποίηση της διάρκειας των μαθημάτων ανά ημέρα.

- **Ισοκατανομή Ωρών Διδασκαλίας ανά Καθηγητή** : αν δεν συμπληρώνεται ο μέγιστος αριθμός ωρών διδασκαλίας ανά Καθηγητή (π.χ με μέγιστο 16 αναλαμβάνει 14) ο στόχος είναι να ισοκατανεμηθούν κατά το δυνατόν οι ώρες στους Καθηγητές έτσι ώστε να μην παρουσιάζεται το φαινόμενο κάποιοι Καθηγητές να αναλαμβάνουν ώρες κοντά στο μέγιστο και κάποιοι άλλοι να κάνουν πολύ λίγες.
- **Ελαχιστοποίηση Ημερών Διδασκαλίας ανά Καθηγητή (optional)** : λαμβάνεται μέριμνα ώστε οι ώρες κάθε Καθηγητή να μαζεύονται στον ελάχιστο δυνατό αριθμό ημερών (π.χ. έκτακτοι Καθηγητές ΤΕΙ)
- **Πολλαπλές ώρες του ίδιου Τμήματος να ανατίθενται στον ίδιο Καθηγητή (optional/constraint)** : αν ένα Τμήμα έχει πολλαπλές ώρες μέσα στην περίοδο (εβδομάδα) π.χ. 2 δίωρα , στόχος είναι οι ώρες αυτές να ανατίθενται στον ίδιο Καθηγητή.
- **Πολλαπλές ώρες του ίδιου Τμήματος να γίνονται την ίδια ώρα της ημέρας (optional/constraint)** : αν ένα Τμήμα έχει πολλαπλές ώρες μέσα στην περίοδο (εβδομάδα) π.χ. 2 δίωρα , στόχος είναι οι ώρες αυτές να γίνονται την ίδια ώρα της ημέρας (π.χ. Δευτέρα 8-10 και Πέμπτη 8-10).
- **Πολλαπλές ώρες του ίδιου Τμήματος να γίνονται στην ίδια αίθουσα (optional/constraint)** : αν ένα Τμήμα έχει πολλαπλές ώρες μέσα στην περίοδο (εβδομάδα) π.χ. 2 δίωρα , στόχος είναι οι ώρες αυτές να γίνονται στην ίδια αίθουσα (π.χ. Δευτέρα 8-10 αίθουσα Α1 και Πέμπτη 8-10 αίθουσα Α1).
- **Ελαχιστοποίηση Χρόνου Μεταφοράς Τμημάτων του ίδιου Εξαμήνου (optional)** : στόχος είναι να ελαχιστοποιηθεί ο χρόνος μεταφοράς των Τμημάτων από μία αίθουσα σε μία άλλη (Travel Time) μεταξύ δύο διαδοχικών μαθημάτων / τμημάτων του ίδιου εξαμήνου.
- **Ελαχιστοποίηση Χρόνου Μεταφοράς Καθηγητών (optional)** : στόχος είναι να ελαχιστοποιηθεί ο χρόνος μεταφοράς των Καθηγητών από μία αίθουσα σε μία άλλη (Travel Time) μεταξύ δύο διαδοχικών μαθημάτων / τμημάτων.

Για όλους τους παραπάνω περιορισμούς και στόχους μπορεί ο χρήστης να καθορίζει δυναμικά έναν συντελεστή «βαρύτητας» για το κατά πόσον ο στόχος / περιορισμός είναι σημαντικός ή όχι. Επίσης μπορεί να θέτει «εκτός» κάποιους περιορισμούς ή στόχους όταν αυτοί δεν υφίστανται ή δεν έχουν νόημα (π.χ. η μετακίνηση καθηγητών και σπουδαστών, όταν όλες οι αίθουσες είναι στο ίδιο κτίριο). Η εξαίρεση ενός περιορισμού / στόχου μπορεί να γίνει εύκολα, βάζοντας συντελεστή βαρύτητας ίσο με το μηδέν.

1.4.4. Οι Γενετικοί Τελεστές που χρησιμοποιήθηκαν

Η κωδικοποίηση του προβλήματος σε δυαδικές συμβολοσειρές διευκόλυνε την χρήση συμβατικού τελεστή Διασταύρωσης (Crossover), για τον ανασυνδυασμό των λύσεων του ΓΑ, καθώς και συμβατικού τελεστή δυαδικής μετάλλαξης (binary mutation). Οι πιθανότητες εφαρμογής των δύο αυτών τελεστών ρυθμίζονται αυτόματα και με δυναμικό τρόπο κατά τη διάρκεια εκτέλεσης του ΓΑ, έτσι ώστε να αποφεύγεται η πρώιμη σύγκλιση καθώς και η υπερβολική διασπορά του πληθυσμού.

Η επιλογή των γονέων για αναπαραγωγή έγινε με την συμβατική μέθοδο του «Τροχού της Ρουλέτας» (Roulette Wheel Parent Selection).

Σε κάθε γενιά ο πληθυσμός των γονέων αντικαθίσταται εξ' ολοκλήρου από τον πληθυσμό των απογόνων. Η καλύτερη όμως λύση κάθε γενιάς μεταφέρεται αυτούσια στην επόμενη γενιά (Elitism).

Εκτός από τους βασικούς γενετικούς τελεστές αναπτύχθηκαν και δοκιμάστηκαν και μία σειρά από ειδικούς τελεστές για συνδυαστικά προβλήματα. ΟΙ προηγμένοι αυτοί τελεστές χωρίζονται σε δύο κατηγορίες : τους τελεστές αναπαραγωγής και τους τελεστές αναρρίχησης.

Τελεστές αναπαραγωγής είναι οι τελεστές που χρησιμοποιούνται συμπληρωματικά ως προς τους βασικούς γενετικούς τελεστές (Crossover και Mutation) και οι οποίοι συμβάλλουν στην παραγωγή απογόνων προκαλώντας ειδικού τύπου μεταλλάξεις στις παραγόμενες λύσεις. Οι τελεστές αυτοί δεν εφαρμόζονται πάντα αλλά με κάποια πιθανότητα και είναι οι ακόλουθοι :

1. Swap Bit Operator
2. Swap Window Operator
3. Swap Chromosome Operator
4. Window Mutation Operator
5. Mutate Chromosome Operator
6. Random Genotype Operator

Η λειτουργία των τελεστών αυτών περιγράφεται στο Κεφάλαιο 2.4.

Τελεστές Αναρρίχησης είναι οι τελεστές που εφαρμόζονται στην καλύτερη λύση κάθε γενιάς, πάντα ή με πιθανότητα, και εκτελούν ουσιαστικά μία μικρής κλίμακας τοπική αναζήτηση (Local Search), στην γειτονιά της καλύτερης λύσης. Εάν κατά την αναζήτηση βρουν ακόμα καλύτερη λύση, τότε αντικαθιστούν την παλιά λύση με τη νέα. Εάν δεν βρουν καλύτερη λύση τότε αφήνουν άθικτη την υπάρχουσα. Οι συγκεκριμένοι τελεστές αναρρίχησης που αναπτύχθηκαν και χρησιμοποιήθηκαν είναι ειδικά κατασκευασμένοι για εφαρμογή σε συνδυαστικά προβλήματα και είναι οι εξής :

1. Bit Swap-Mutate Hill Climbing Operator
2. Window Swap Hill Climbing Operator
3. Day Climb Operator
4. Fix Teacher Hill Climbing Operator
5. Fix Room Hill Climbing Operator
6. Micro GA Combinatorial Hill Climbing Operator

Η λειτουργία των τελεστών αυτών επίσης περιγράφεται στο Κεφάλαιο 2.4.

Ως όριο τερματισμού του ΓΑ χρησιμοποιείται το απόλυτο όριο γενεών, το οποίο στο βασικό σετ παραμέτρων είχε την τιμή 5000.

Για τον χειρισμό των περιορισμών που συναντώνται στα προβλήματα Ωρολόγιου Προγράμματος χρησιμοποιήσαμε μία καινοτόμο τεχνική την τεχνική της Μεταβλητής Συνάρτησης Ποιότητας (Varying Fitness Function). Η ιδέα πίσω από την τεχνική έγκειται στο ότι η ιδανικότερη περίπτωση αντιμετώπισης των περιορισμών θα ήταν να διατηρούνται οι συντελεστές ποινής σε χαμηλά επίπεδα κατά την διάρκεια που ο Γ.Α. ανιχνεύει τον χώρο λύσεων για τον εντοπισμό της γενικής περιοχής λύσεων υψηλής ποιότητας, έτσι ώστε να μπορεί να εκμεταλλεύεται την πληροφορία των μη αποδεκτών λύσεων γεφυρώνοντας τα χάσματα ανάμεσα σε γειτονικές μη συνεκτικές περιοχές αποδεκτών λύσεων. Μετά όμως από τον εντοπισμό των περιοχών υψηλής ποιότητας θα έπρεπε η Συνάρτηση Ποινής να γίνεται πιο αυστηρή ώστε να διαχωρίζονται αποδεκτές και μη αποδεκτές λύσεις και να αποκλείεται η περίπτωση σύγκλισης σε λύσεις που δεν είναι έγκυρες. Έτσι καταλήγουμε σε μία συνάρτηση ποινής που εξαρτάται όχι μόνο από την παραβίαση των περιορισμών αλλά και από τον χρόνο που μπορεί να εκφραστεί μέσω του δείκτη γενιάς g.

$$Qv(\Lambda,g) = O(\Lambda) + Pv(\Lambda,g)$$

Η συνάρτηση ποινής θα πρέπει επομένως να είναι μονότονα αύξουσα ως προς Λ και ως προς g. Επίσης θα πρέπει να ισχύουν τα εξής :

$Pv(\Lambda,g) \rightarrow 0$ όταν $g \rightarrow 0$ και
$Pv(\Lambda,g) \rightarrow P(\Lambda)$ όταν $g \rightarrow G$

όπου $P(\Lambda)$ η μη μεταβλητή συνάρτηση ποινής κατάλληλα φτιαγμένη ώστε να διαχωρίζει αποδεκτές και μη αποδεκτές λύσεις, και G το όριο γενεών του Γ.Α.

Υπάρχει πληθώρα συναρτήσεων που να ικανοποιούν τις παραπάνω προδιαγραφές. Μία απλή περίπτωση είναι η Γραμμική Μεταβλητή Συνάρτηση Ποινής που έχει την μορφή :

$$Pv(\Lambda,g) = \frac{g}{G} . P(\Lambda) =$$

Γενικά μία Μεταβλητή Συνάρτηση Ποινής μπορεί να εκφραστεί ως το γινόμενο της σταθερής Συνάρτησης Ποινής επί μία αύξουσα συνάρτηση του δείκτη γενιάς g, με κατώτατη τιμή το 0 και ανώτατη το 1, δηλαδή :

$Pv(\Lambda,g) = V(g) . P(\Lambda)$ όπου
$V(g) \rightarrow 0$ για $g \rightarrow 0$ και
$V(g) \rightarrow 1$ για $g \rightarrow G$

Η συνάρτηση V(g) μπορεί να είναι οποιαδήποτε αύξουσα συνάρτηση αρκεί να πληροί τις πιο πάνω προδιαγραφές. Έτσι μπορεί να είναι γραμμική, τετραγωνική, εκθετική ή και βηματική. Η συνάρτηση V(g) καθορίζει στην ουσία την στρατηγική μεταβολής των ποινών και είναι καθοριστικής σημασίας, όπως αποδεικνύεται από τα πειράματα για την απόδοση του Γ.Α.

Η μεταβλητή συνάρτηση ποινής δημιουργεί ένα μεταβλητό χώρο λύσεων του προβλήματος. Αυτό σημαίνει ότι οι ίδιες λύσεις σε διαδοχικές γενιές μπορεί να μην έχουν την ίδια τιμή ποιότητας (αν οι λύσεις αυτές παραβιάζουν κάποιους περιορισμούς). Αυτό πιθανόν να ήταν μειονέκτημα για άλλες μεθόδους, αλλά ο Γ.Α., λόγω ακριβώς της ικανότητάς του να προσαρμόζει τον πληθυσμό ανάλογα με τον εκάστοτε χώρο λύσεων που δημιουργεί η Συνάρτηση Ποιότητας, δεν έχει ανάλογο πρόβλημα. Οι Γ.Α. όμως, λόγω του πληθυσμού λύσεων που διατηρούν έχουν κάποια αδράνεια Μ στην παρακολούθηση των αλλαγών του χώρου λύσεων. Λόγω της αδράνειας αυτής που εξαρτάται τόσο από το μέγεθος του πληθυσμού όσο και από τις πιθανότητες εφαρμογής καθώς και τις ιδιαιτερότητες των γενετικών τελεστών αλλά και από την μέθοδο επιλογής γονέων, οι Γ.Α. έχουν κάποιο όριο στην ταχύτητα προσαρμογής τους. Έτσι όταν η συνάρτηση V(g) μεταβάλλει πολύ γρήγορα τον χώρο λύσεων τότε υπάρχει περίπτωση ο Γ.Α. να μην είναι σε θέση να προσαρμοστεί στις αλλαγές αυτές. Έτσι αν για παράδειγμα η συνάρτηση V(g) μεταβάλλεται εξαιρετικά γρήγορα κατά το τέλος του Γ.Α. υπάρχει περίπτωση ο τελευταίος να μην μπορέσει να προσαρμοστεί στις αυξανόμενες τιμές Ποινών και έτσι να μην προλάβει να απομακρυνθεί από τυχόν μη αποδεκτές λύσεις (καλής όμως ποιότητας όσον αφορά την αντικειμενική συνάρτηση) και να συγκλίνει τελικά σε αυτές.

Επίσης, αν η συνάρτηση V(g) αυξάνεται πολύ γρήγορα στην αρχή του Γ.Α. τότε ο τελικός χώρος λύσεων με τις υψηλές τιμές ποινών διαμορφώνεται σχετικά γρήγορα, οπότε ο Γ.Α. προσαρμόζεται απευθείας στην τελική μορφή του χώρου λύσεων οπότε και δεν υπάρχει κίνδυνος σύγκλισης σε μη αποδεκτές λύσεις. Έτσι όμως καταστρέφεται το πλεονέκτημα της μεθόδου να διατηρεί στην αρχή τις ποινές χαμηλά ώστε ο Γ.Α. να μπορέσει να εκμεταλλευτεί την πληροφορία των μη αποδεκτών λύσεων για να οδηγηθεί προς το βέλτιστο.
Από τα παραπάνω συμπεραίνουμε ότι η μορφή της συνάρτησης V(g) πρέπει να επηρεάζει ισχυρά την απόδοση του Γ.Α. ιδιαίτερα στην περίπτωση περίπλοκων περιορισμών που δημιουργούν μη κυρτούς και μη συνεκτικούς χώρους λύσεων.

1.5 Βιβλιογραφία

[Cart86] Michael W. Carter. “A survey of practical applications of examination timetabling algorithms”. Operations Research, 34(2):193--202, March-April 1986.

[CoDM90] A. Colorni, M. Dorigo, and V. Maniezzo. “A genetic algorithm to solve the timetable problem”. Technical Report 90-060, Politecnico di Milano, 1990. submitted to Computational Optimization and Applications Journal.

[AbAb91] D. Abramson and J. Abela. “A parallel genetic algorithm for solving the school timetabling problem”. Technical report, Division of Information Technology, C.S.I.R.O., April 1991.

[CoFM93] D. Corne, H. L. Fang, and C. Mellish. Solving the modular exam scheduling problem with genetic algorithms. In Proceedings of the Sixth International Conference on Industrial and Engineering Aplications of Artificial Intelligence and Expert Systems, pages 370--373, 1993.

[CoRF94] D. Corne, P. Ross, and H-L. Fang. Fast practical evolutionary timetabling. Research Report 708, Dept. of Artificial Intelligence, University of Edinburgh, 1994. To appear in Proc. of the AISB Workshop on Evolutionary Computation, Springer-Verlag, 1994.

[PaLCP94] B.Paechter, H.Luchian, A.Cumming, and M.Petruic. “Two solutions to the general timetable problem using evolutionary methods”. In Proceedings of the First IEEE Conference on Evolutionary Computation, pages 300--305, 1994.

[BuEW94] E. K. Burke, D. G. Elliman, and R. F. Weare. “A genetic algorithm based university timetabling system”. In Proceedings of the 2nd East-West International Conference on Computer Technologies in Education (Crimea, Ukraine, 19th-23rd Sept 1994), volume 1, pages 35--40, 1994.

[BuEW94B] Edmund Burke, David Elliman, and Rupert Weare. “A genetic algorithm for university timetabling”. In AISB Workshop on Evolutionary Computation. Workshop Notes, 1994.

[Chan94] Yam Ling Chan. “A genetic algorithm (GA) shell for iterative timetabling”. Master's thesis, Depart- ment of Computer Science, RMIT, 1994.

[BuEW95] E. K. Burke, D. G. Elliman, and R. F. Weare. “The automated timetabling of university exams using a hybrid genetic algorithm”. In AISB Workshop on Evolutionary Computing (University of Leeds, UK, 3-7 April 1995), Society for the Study of Artificial Intelligence and Simulation of Behaviour (SSAISB), 1995.

[BuEW95B] E. K. Burke, D. G. Elliman, and R. F. Weare. “A hybrid genetic algorithm for highly constrained timetabling problems”. In 6th Intenational Conference on Genetic Algorithms (Pittsburgh, USA), 1995.

[CaLa95] Michael W. Carter and Gilber Laporte. “Recent developments in practical exam timetabling”. In E.K. Burke and P.M. Ross, editors. Proceedings of the 1st International Conference on the Practice and Theory of Automated Timetabling, 1995.

[BuNW95] E.K. Burke, J.P.Newall, and R.F.Weare. “A memetic algorithm for university exam timetabling”. In E.K. Burke and P.M. Ross, editors. Proceedings of the 1st International Conference on the Practice and Theory of Automated Timetabling, 1995.

[Erbe95] W. Erben. “Timetabling using genetic algorithms”. In Artificial Neural Nets and Genetic Algorithms, Proceedings of the International Conference in Ales (France), pages 30--32, 1995.

[ErKe95] Wilhelm Erben and Jurgen Keppler. “A genetic algorithm solving an weekly course-timetabling problem”. In Proceedings of the First International Conference on the Practice and Theory of Automated Timetabling (ICPTAT '95), pages 21--32, 1995.

[MaRe97] Nuno Mamede and Tiago Rente. “Repairing timetables using genetic algorithms and simulated annealing”. In E.K. Burke and M. Carter, editors. Proceedings of the 2nd International Conference on the Practice and Theory of Automated Timetabling, 1997.

[CaLa97] Michael W. Carter and Gilber Laporte. “Recent developments in practical course timetabling”. In . E.K. Burke and M. Carter, editors. Proceedings of the 2nd International Conference on the Practice and Theory of Automated Timetabling, 1997.

[PaRC97] Ben Paechter, R C Rankin, and Andrew Cumming. “Improving a lecture timetabling system for university-wide use”. In In E.K. Burke and M. Carter, editors. Proceedings of the 2nd International Conference on the Practice and Theory of Automated Timetabling, 1997.

[BuCa97] E.K. Burke and M. Carter, editors. Proceedings of the 2nd International Conference on the Practice and Theory of Automated Timetabling, 1997.

[CaAR97] J P Caldeira and Agostinho C Rosa. School timetabling using genetic search. In E.K. Burke and M. Carter, editors. Proceedings of the 2nd International Conference on the Practice and Theory of Automated Timetabling, 1997.

[Corne97] D. Corne. “Evolutionary approaches to the partition/timetabling problem”. In Proceedings of IC- ANNGA'97 -- The International Conference on Artificial Neural Networks and Genetic Algorithms. Springer, 1997. to appear.

[AdAr99] Adamidis P., Arapakis P., “Evolutionary Algorithms in Lecture Timetabling”, *Congress on Evolutionary Computation*, July 6-9, 1999, Washington D.C., USA

Κεφάλαιο 2. Βέλτιστη ρύθμιση των παραμέτρων του αλγορίθμου με βάση πειραματικά δεδομένα επί πραγματικών περιπτώσεων

2.1 Εισαγωγή

Για την πιστοποίηση της αποτελεσματικότητας του ΓΑ στην επίλυση προβλημάτων Ωρολόγιου Προγράμματος απαιτούνται δύο (2) βήματα :

A. Πιλοτική εφαρμογή του ΓΑ πάνω σε πραγματικό πρόβλημα Ωρολόγιου Προγράμματος και ,
B. Εκτενείς δοκιμές για διάφορες τιμές των παραμέτρων του ΓΑ για την βέλτιστη ρύθμισή τους και την μεγιστοποίηση της απόδοσης της μεθόδου

Το πρώτο από αυτά απαιτεί τη συγκέντρωση των στοιχείων ενός πραγματικού Ωρολόγιου Προγράμματος, που θα αποτελέσει την πλατφόρμα δοκιμών του ΓΑ. Για το σκοπό αυτό χρησιμοποιήθηκε το Ωρολόγιο Πρόγραμμα Σπουδών του Τμήματος Πληροφορικής & Επικοινωνιών του ΤΕΙ Σερρών, για το Εαρινό εξάμηνο του 2004. Με βάση το πρόγραμμα αυτό θα πρέπει να δημιουργηθεί ένα αρχείο διάρθρωσης ("setup file") το οποίο κωδικοποιεί τα δεδομένα τα οποία λαμβάνει από το ωρολόγιο πρόγραμμα σε μορφή τέτοια ώστε να μπορεί η εφαρμογή Βελτιστοποίησης Ωρολογίου Προγράμματος με Γενετικό Αλγόριθμο να τα διαβάζει και να τα επεξεργάζεται.

Το δεύτερο βήμα απαιτεί τη εύρεση εκείνου του συνόλου παραμέτρων του ΓΑ οι οποίες μεγιστοποιούν την απόδοση της μεθόδου, εκείνες δηλαδή που επιτυγχάνουν την παραγωγή λύσεων πολύ κοντά στο βέλτιστο, με παράλληλη ικανοποίηση όλων των περιορισμών, με συνέπεια και αξιοπιστία.

Καθένα από τα παραπάνω βήματα παρουσιάζεται εκτενέστερα στις ενότητες που ακολουθούν.

2.2 Η κωδικοποίηση ενός προβλήματος Time Tabling

Δοθέντος του προγράμματος σπουδών ενός τμήματος αλλά και της τρέχουσας πληροφορίας σχετικά με το πλήθος των «πόρων» (Μαθήματα, Καθηγητές, Αίθουσες κ.λ.π.) που συμμετέχουν σε αυτό, το αρχείο διάρθρωσης περιέχει αρχικά κάποιες πληροφορίες και στη συνέχεια βάσει αυτών των πληροφοριών κωδικοποιεί το ωρολόγιο πρόγραμμα.

Οι έννοιες οι οποίες χρησιμοποιούνται για την κωδικοποίηση είναι οι ακόλουθες:

(Α) **Μάθημα** (**Lesson)**: που αντιστοιχεί στο πλήθος των διαφορετικών μαθημάτων τα οποία διδάσκονται και τα οποία μπορούν να προκύψουν από τον οδηγό - πρόγραμμα σπουδών. Αξίζει να σημειωθεί ότι εφόσον ένα μάθημα πραγματοποιείται τόσο σε επίπεδο θεωρίας όσο και σε επίπεδο εργαστηρίου εκλαμβάνεται ως δυο διαφορετικά μαθήματα. Η έννοια του μαθήματος (Lesson) χρησιμοποιείται για την κωδικοποίηση όλων των μαθημάτων που περιέχονται στο πρόγραμμα σπουδών.

(Β) **Τάξη** (**Class):** που αντιστοιχεί στον αριθμό των μαθημάτων όλων των εξαμήνων και προκύπτει από την έννοια του μαθήματος (Lesson). Για παράδειγμα η έννοια της Τάξης (Class) ταυτίζεται με αυτή του μαθήματος (Lesson) στην περίπτωση ενός θεωρητικού μαθήματος, αλλά όχι απόλυτα στην περίπτωση ενός εργαστηριακού μαθήματος. Στην τελευταία περίπτωση, αν Χ είναι το πλήθος των τμημάτων τα οποία πραγματοποιούν το ίδιο εργαστηριακό μάθημα σύμφωνα με τις ανάγκες και το πλήθος των φοιτητών που το παρακολουθούν, τότε για το εν λόγω μαθήματος (Lesson) θα δημιουργηθούν Χ διαφορετικές τάξεις (Classes) (π.χ. αν στο εργαστήριο Δομημένου Προγραμματισμού υπάρχουν πέντε

τμήματα τότε παρόλο που θα αναφερόμαστε σε ένα μάθημα (Lesson) θα δημιουργηθούν πέντε διαφορετικές τάξεις (Classes) στο ίδιο αρχείο κωδικοποίησης.
(C) **Αίθουσες (Rooms)**: το οποίο αντιστοιχεί στο πλήθος των διαθέσιμων αιθουσών είτε σε αυτές πραγματοποιούνται μαθήματα θεωρίας είτε εργαστηριακές ασκήσεις.
(D)**Καθηγητές (Teachers)**: το οποίο αντιστοιχεί στο πλήθος των καθηγητών (μονίμων και εκτάκτων) που διδάσκουν τόσο τα μαθήματα της θεωρίας όσο και τα εργαστηριακά μαθήματα. Αξίζει να σημειωθεί ότι, κατά τη διάρκεια μοντελοποίησης του προβλήματος για το ωρολόγιο πρόγραμμα του τμήματος Πληροφορικής και Επικοινωνιών του ΤΕΙ Σερρών, σε ότι αφορά τους μόνιμους καθηγητές ήταν γνωστή η πληροφορία τόσο του πλήθους των διδασκόντων όσο και των ωρών διδασκαλίας καθενός εξ' αυτών. Σε ότι αφορά τους έκτακτους καθηγητές τόσο η πληροφορία του πλήθους τους όσο και η πληροφορία των ωρών τις οποίες καθένας από αυτούς δίδασκε έπρεπε να προκύψουν από το υπάρχον ωρολόγιο πρόγραμμα σπουδών.
(E) **Ημέρες (Days)**: που ισούται με πέντε και αντιστοιχεί στο πλήθος των ημερών μέσα στην εβδομάδα που διεξάγονται μαθήματα.
(F) **Μέγιστος αριθμός ωρών ανά ημέρα (Max Hours Per Day)**: που ισούται με 13 και αντιστοιχεί στο πλήθος των ωρών μέσα στην ημέρα που μπορούν να διεξαχθούν μαθήματα, κάθε μια από τις οποίες συνιστά μια περίοδο. Πιο συγκεκριμένα η διδασκαλία των μαθημάτων πραγματοποιείται κατά τη διάρκεια μιας ημέρας από τις 8.00 π.μ. ως 21.00 μ.μ.
(G) **Χρονική Περίοδος (Period Time)**: που ισούται με 60 και αντιστοιχεί στο πλήθος των λεπτών που συνιστούν μια περίοδο.
(H) **Παύση (Break Time)**: που αντιστοιχεί στο χρονικό διάστημα ή αλλιώς παύση - διάλειμμα ανάμεσα σε δυο περιόδους και το οποίο συμπεριλαμβάνεται στο χρονικό διάστημα μιας περιόδου.
(I) **Πλήθος τύπων τάξεων (No of Class Types)**: που ισούται με δύο και αντιστοιχεί στον τύπο των μαθημάτων τα οποία έχουμε. Στο ωρολόγιο πρόγραμμα του Τμήματος Πληροφορικής και Επικοινωνιών του ΤΕΙ Σερρών έχουμε δυο τύπους μαθημάτων: τις θεωρίες και τα εργαστήρια.
(K) **Εβδομαδιαίο Πρόγραμμα Τάξεων (Class Week Schedule):** που περιέχει για κάθε ένα από τα μαθήματα τύπου Class μια γραμμή της μορφής:
Class A Semester B Lesson C TYPE D : E F ……
Όπου A είναι ο αύξων αριθμός του Μαθήματος
B = το εξάμηνο στο οποίο διδάσκεται η εν λόγω τάξη (Class)
C = το μάθημα (Lesson) στο οποίο αντιστοιχεί η εν λόγω τάξη (Class)
D = ο τύπος του μαθήματος (π.χ. αν είναι θεωρητικό ή εργαστηριακό)
E, F = οι αριθμοί οι οποίοι αντιστοιχούν στις ώρες που διδάσκεται αυτό το μάθημα π.χ. αν E=2 και F= 0 τότε το μάθημα έχει συνολικά δυο ώρες οι οποίες διδάσκονται αναπόσπαστα ενώ αν π.χ. αν E=2 και F= 1 τότε το μάθημα διδάσκεται συνολικά τρεις ώρες υπό τη μορφή ενός δίωρου και ενός ωριαίου μαθήματος.
(L) **Ανταγωνιστικότητα μεταξύ διαφορετικών τύπων (Inter Type Concurrency):** Στη συνέχεια παραθέτεται η ανταγωνιστικότητα μεταξύ διαφορετικών τύπων μαθημάτων. Αυτό σημαίνει ότι σε επίπεδο εξαμήνου κατά πόσο μπορούν στο πρόγραμμα να διδάσκονται την ίδια μέρα και ώρα ταυτόχρονα δυο μαθήματα θεωρίας, δυο εργαστηριακά μαθήματα ή ένα μάθημα θεωρίας και ένα εργαστηριακό μάθημα. Η τιμή 0 δηλώνει ασυμβατότητα ανάμεσα σε δυο τύπους ενώ η τιμή 1 συμβατότητα. Για τα προβλήματα τα οποία εξετάζουμε θεωρούμε ότι η μοναδική συμβατότητα η οποία μπορεί να είναι αποδεκτή είναι η ταυτόχρονη διεξαγωγή εργαστηριακών μαθημάτων είτε αυτά αντιστοιχούν στο ίδιο είτε σε διαφορετικά μαθήματα (Lessons). Αυτό οφείλεται στο γεγονός ότι σε ένα μάθημα (Lesson) το οποίο αντιστοιχεί σε ένα εργαστηριακό μάθημα αντιστοιχούν περισσότερα από μια τάξη (Class) επομένως είναι

δυνατό ένας φοιτητής να εγγραφεί σε τάξη (Class) μαθημάτων τα οποία **δεν** έχουν αλληλοσυγκρουόμενες ώρες.

(M) **Ελάχιστη απόσταση σε ημέρες μεταξύ τάξεων με πολλαπλές περιόδους (Minimum Distance (days) of MultiPeriod Classes)**: το οποίο περιλαμβάνει καταχωρήσεις της μορφής Class X: Y όπου το X αντιστοιχεί στον αύξοντα αριθμό της τάξης (Class) και το Y στο πλήθος των ημερών που πρέπει να παρεμβάλλονται ανάμεσα σε διαφορετικές διαλέξεις του ιδίου μαθήματος. Στην περίπτωση του προγράμματος σπουδών του Τμήματος Πληροφορικής και Επικοινωνιών του ΤΕΙ Σερρών το Y έλαβε την τιμή 1.

(N) **Πλήθος Καθηγητών ανά Τάξη (No of Teachers per Class)**: το οποίο περιλαμβάνει καταχωρήσεις της μορφής (για κάθε μια από τις υπάρχουσες τάξεις (Class)) Class X: Y όπου X είναι ο αύξων αριθμός της τάξης (Class) και Y το πλήθος των διδασκόντων καθηγητών. Κάνουμε εδώ τη σύμβαση ότι το πλήθος των διδασκόντων στα εργαστηριακά μαθήματα είναι δυο και στα μαθήματα της θεωρίας ένα, άσχετα με το κατά πόσο αυτό είναι εφικτό λόγω άλλων περιορισμών όπως π.χ. το μέγιστο αριθμό των ωρών που μπορεί να έχει ένας διδάσκοντας.

(O) **Συσχέτιση μεταξύ Τάξεων και Αιθουσών** (**Class / Room Relationships)**: το οποίο περιλαμβάνει για κάθε μια από τις τάξεις (Class) μια καταχώρηση της μορφής: Class X: A B C D ….. η οποία δηλώνει σε ποιες αίθουσες (A, B, C) μπορεί να διεξαχθεί η τάξη - Class X.

Αξίζει να σημειωθεί ότι στο ωρολόγιο πρόγραμμα του Τμήματος Πληροφορικής και Επικοινωνιών του ΤΕΙ Σερρών κάναμε τη σύμβαση ότι ένα μάθημα είναι δυνατό να διδαχθεί σε οποιαδήποτε αίθουσα - αμφιθέατρο αλλά όχι εργαστήριο η οποία προσφέρεται για διδασκαλία. Αντίθετα, σε ότι αφορά τα εργαστήρια, η φύση της τάξης (Class) υποδεικνύει και το πλήθος των εργαστηρίων στα οποία μπορεί να διεξαχθεί, στοιχείο το οποίο είναι άμεσα συνυφασμένο με τον εξοπλισμό του εκάστοτε εργαστηρίου.

(P) **Σχέση μεταξύ Αιθουσών και Ωρών (Room/ Hour Relationships)**: το οποίο περιλαμβάνει για κάθε μια από τις διαθέσιμες αίθουσες καταχωρήσεις της μορφής:

Room A
Day 1 : C D E … K
Day 2 : C D E … K
Day 3 : C D E … K
Day 4 : C D E … K
Day 5 : C D E … K

Όπου A είναι ο αύξων αριθμός της αίθουσας.

Η γραμμής Day 1 : C D E … K

Δηλώνει την πρώτη μέρα (Δευτέρα) και C D E … K είναι αύξοντες αριθμοί οι οποίοι αντιστοιχούν στις διδακτικές ώρες. Μια καταχώρηση της μορφής Day 1 : 1 2 3 4 …. 13 δηλώνει ότι η εν λόγω αίθουσα είναι διαθέσιμη και τις 13 διδακτικές ώρες της ημέρας.

(Q) Απόσταση μεταξύ Αιθουσών (Room-to-Room Distance): η οποία παρέχει την πληροφορία της απόστασης σε λεπτά που απαιτούνται για την μετάβαση από τη μια αίθουσα σε μια άλλη. Η εν λόγω πληροφορία παριστάνεται με τη βοήθεια ενός άνω ή κάτω τριγωνικού πίνακα. Πιο συγκεκριμένα μια τέτοια πληροφορία έχει τη μορφή:

Rooms	1	2	3
1	0		
2	X	0	
3	Y	X	0

όπου X και Y είναι αποστάσεις σε λεπτά για την μετάβαση από τη μια αίθουσα στην άλλη.

(**R**) **Σχέσεις μεταξύ Καθηγητών και Τάξεων (Teacher / Class Relationships):** το οποίο περιλαμβάνει την πληροφορία των μαθημάτων τα οποία διδάσκει καθένας από τους καθηγητές. Μια καταχώρηση της μορφής:

Teacher X: A B C D E F δηλώνει ότι ο καθηγητής X διδάσκει τις τάξεις (Class) A, B, C, D, E, και F. Στην περίπτωση που μια τάξη (class) διδάσκεται αποκλειστικά και μόνο από τον καθηγητή X τότε βάζουμε μπροστά στον αριθμό της τάξης (class) το πρόσημο μείον.

Στην περίπτωση του προγράμματος του Τμήματος Πληροφορικής και Επικοινωνιών του ΤΕΙ Σερρών, για όλες τις τάξεις (class) οι οποίες αντιστοιχούν σε θεωρητικά μαθήματα τοποθετούμε το πρόσημο μείον. Το ίδιο πρόσημο τοποθετείται και σε εκείνα τις τάξεις (class) που αντιστοιχούν σε εργαστηριακά μαθήματα του ιδίου μαθήματος (Lesson) και τα οποία πραγματοποιούνται **πάντα** από τον ίδιο καθηγητή. Στην περίπτωση που οι τάξεις (Class) του ιδίου μαθήματος (Lesson) πραγματοποιούνται από πολλούς καθηγητές, όπου κανένας από αυτούς δεν συμμετέχει σε όλα τις τάξεις (Class) δεν βάζουμε το πρόσημο μείον.

(S) **Σχέσεις μεταξύ Καθηγητών και Ωρών (Teacher/ Hour Relationships):** στη συνέχεια ακολουθεί η δήλωση της διαθεσιμότητας των καθηγητών σε όλη τη διάρκεια της εβδομάδας. Για κάθε έναν από τους διδάσκοντες καθηγητές υπάρχει μια καταχώρηση της μορφής :

Teacher X Max Hours = Y Min Hours = Z
Day 1: 1 2 3 4 5 6 7 8 9 10 11 12 13
Day 2: 1 2 3 4 5 6 7 8 9 10 11 12 13
Day 3: 1 2 3 4 5 6 7 8 9 10 11 12 13
Day 4: 1 2 3 4 5 6 7 8 9 10 11 12 13
Day 5: 1 2 3 4 5 6 7 8 9 10 11 12 13

Όπου X είναι ο αύξων αριθμός του εν λόγω μαθήματος, Y είναι το μέγιστο πλήθος ωρών που μπορεί να διδάξει (ανάλογα με τη βαθμίδα στην οποία ανήκει και το σύνολο των υποχρεώσεων του στο τμήμα) και Z είναι το πλήθος των ελάχιστων ορών που μπορεί να διδάξει. Στη συνέχεια για κάθε μια από τις ημέρες τοποθετούνται οι ώρες (από τις 13 διδακτικές που αντιστοιχούν στις ώρες διδασκαλίας μιας ημέρα) κατά τις οποίες είναι διαθέσιμος ο εκάστοτε καθηγητής.

2.3 Κωδικοποίηση πραγματικού προβλήματος ωρολόγιου προγράμματος του ΤΕΙ Σερρών

Για την πιλοτική εφαρμογή του αναπτυχθέντος λογισμικού αυτόματης και βέλτιστης παραγωγής ωρολόγιων προγραμμάτων επιλέχθηκε η πραγματική περίπτωση ωρολόγιου προγράμματος του Τμήματος Πληροφορικής και Επικοινωνιών του ΤΕΙ Σερρών. Οι λόγοι που μας οδήγησαν στην επιλογή της συγκεκριμένης περίπτωσης προγράμματος ήταν αφενός μεν η εγγενής δυσκολία στην κατάστρωσή του λόγω πολλών και διαφορετικού τύπου περιορισμών που τίθενται, και αφετέρου το γεγονός ότι έχοντας και το πραγματικό πρόγραμμα που αναπτύχθηκε από τους καθηγητές του ΤΕΙ Σερρών μπορούσαμε να συγκρίνουμε άμεσα την απόδοση του ΓΑ.

Το συγκεκριμένο πρόγραμμα έχει τα εξής στοιχεία :

Μαθήματα (events) που πρέπει να προγραμματιστούν	: 192
Αίθουσες διεξαγωγής μαθημάτων	: 17
Καθηγητές	: 82
Ημέρες διεξαγωγής μαθημάτων	: 5 (πενθήμερη εβδομάδα)
Μέγιστο πλήθος ωρών ανά ημέρα	: 13
Αριθμός διαφορετικών εξαμήνων	: 7
Είδη μαθημάτων	: 2 (Θεωρία–Εργαστήριο)

Το συγκεκριμένο πρόγραμμα επιλέχθηκε λόγω της εγγενούς δυσκολίας που έχει στην κατάστρωσή του που οφείλεται στους εξής παράγοντες :

1. Ο Αριθμός Αιθουσών είναι περιορισμένος
2. Οι αίθουσες (κυρίως των εργαστηρίων) απασχολούνται και τις 13 ώρες της ημέρας συνεχόμενα με ελάχιστα κενά.
3. Έχει μεγάλο αριθμό events που πρέπει να χρονοπρογραμματιστούν, λόγω των πολλών τμημάτων εργαστηρίων.
4. Έχει και μόνιμους καθηγητές που έχουν συγκεκριμένα μαθήματα αλλά και έκτακτους καθηγητές που μπορούν να πάρουν διαφορετικά μαθήματα κάθε φορά.
5. Υπάρχουν πολλοί περιορισμοί ως προς την διεξαγωγή συγκεκριμένων μαθημάτων σε συγκεκριμένες αίθουσες.
6. Υπάρχουν πολλοί και διαφορετικοί (diverse) περιορισμοί ως προς τις ώρες που πρέπει να ανατεθούν σε κάθε καθηγητή.

Στην περίπτωση του ωρολογίου προγράμματος του Τμήματος Πληροφορικής και Επικοινωνιών του ΤΕΙ Σερρών ένας καθηγητής εφαρμογών δεν μπορεί να διδάξει περισσότερες από 18 ώρες και λιγότερες από 15, ένας επίκουρος καθηγητής δεν μπορεί να διδάξει περισσότερες από 15 ώρες και λιγότερες από 14, ένας αναπληρωτής καθηγητής δεν μπορεί να διδάξει περισσότερες από 14 ώρες και λιγότερες από 12, ενώ ένας αναπληρωτής καθηγητής ο οποίος είναι ταυτόχρονα και προϊστάμενος τμήματος δεν μπορεί να διδάξει περισσότερες από 8 ώρες και λιγότερες από 6. Αξίζει να σημειωθεί ότι σε ότι αφορά τους έκτακτους καθηγητές, δεν μπορούν να διδάξουν περισσότερες από 16 ώρες. Πρακτικά, στο παραπάνω ωρολόγιο πρόγραμμα υπολογίσαμε πόσες ώρες δίδασκε ο κάθε έκτακτος καθηγητής – έστω Χ- και θεωρήσαμε ότι οι ελάχιστες ώρες που μπορούσε να διδάξει ήταν Χ ενώ οι μέγιστες Χ+3 εφόσον το Χ+ 3 δεν υπερέβαινε τον αριθμό 16.

Στη συνέχεια παρατίθεται τμήμα του Αρχείου Διάρθρωσης όπως συντάχθηκε και εν συνεχεία χρησιμοποιήθηκε στις δοκιμές. Το αρχείο αυτό περιλαμβάνει όλες τις παραμέτρους και τους περιορισμούς που τίθενται για την κατάστρωση του ωρολόγιου προγράμματος :

```
Classes= 187
Rooms= 17
Teachers= 82
Days= 5
MaxHoursPerDay= 13
PeriodTime= 60
BreakTime= 15
NoOfClassTypes= 2
---- Class Week Schedule
Class   1  Semester 1  Lesson 1  Type A : 3 2
Class   2  Semester 1  Lesson 2  Type A : 3
Class   3  Semester 1  Lesson 3  Type B : 2
Class   4  Semester 1  Lesson 3  Type B : 2
Class   5  Semester 1  Lesson 3  Type B : 2
...
Class 187  Semester 7  Lesson 71  Type B : 2 2
---- Inter type concurrency
Types  A  B
A      0  0
B      0  1
---- Minimum Distance (days) of Multi Period Classes
Class   1 : 1
Class   2 : 1
Class   3 : 1
Class   4 : 1
Class   5 : 1
...
Class 187 : 0
---- No of Teachers per Class
Class  1 : 1
```

```
Class  2 : 1
Class  3 : 2
Class  4 : 2
Class  5 : 2
...
Class 187 : 1
---- Class/Room Relationships
Class 1   :                   10 11 12 13 14 15 16 17
Class 2   :                   10 11 12 13 14 15 16 17
Class 3   :           6
Class 4   :           6
...
Class 187 : 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
---- Room/Hour Relationships
Room 1 :
       Day 1 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 2 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 3 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 4 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 5 : 1 2 3 4 5 6 7 8 9 10 11 12 13
Room 2 :
       Day 1 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 2 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 3 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 4 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 5 : 1 2 3 4 5 6 7 8 9 10 11 12 13
...
Room 17 :
       Day 1 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 2 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 3 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 4 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 5 : 1 2 3 4 5 6 7 8 9 10 11 12 13
---- Room To Room Distance
Rooms 1   2   3   4   5   6   7   8   9   10   11   12   13  14  15  16  17
1     0
2    10   0
3    10  10   0
4    10  10  10   0
5    10  10  10  10   0
6    10  10  10  10  10   0
7    10  10  10  10  10  10   0
8    10  10  10  10  10  10  10   0
9    10  10  10  10  10  10  10  10   0
10   10  10  10  10  10  10  10  10  10   0
11   10  10  10  10  10  10  10  10  10  10   0
12   10  10  10  10  10  10  10  10  10  10  10   0
13   10  10  10  10  10  10  10  10  10  10  10  10   0
14   10  10  10  10  10  10  10  10  10  10  10  10  10   0
15   10  10  10  10  10  10  10  10  10  10  10  10  10  10   0
16   10  10  10  10  10  10  10  10  10  10  10  10  10  10  10   0
17   10  10  10  10  10  10  10  10  10  10  10  10  10  10  10  10   0
---- Teacher/Class Relationships (Minus means fixed assignment)
Teacher 1  :   47   48   49   50   51   52   62   63   64   65   66   67   83   84
85   86   87   88
Teacher 2  :  -24   25   26   27   28   29   30
Teacher 3  :  104  105  106  107  108  109  161  162  163  164  165
Teacher 4  :   55   56   57   58   59   60
Teacher 5  :   18   19   20   21   22   23  -55  -56  -57  -58  -59 -60
...
Teacher 82 :   83   84   85   86   87   88
---- Teacher/Hour Relationships
Teacher  1 : MaxHours= 15  MinHours= 15
       Day 1 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 2 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 3 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 4 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 5 : 1 2 3 4 5 6 7 8 9 10 11 12 13
```

```
Teacher  2 : MaxHours= 7   MinHours= 2
       Day 1 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 2 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 3 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 4 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 5 : 1 2 3 4 5 6 7 8 9 10 11 12 13
...
Teacher 82 : MaxHours= 6   MinHours= 3
       Day 1 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 2 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 3 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 4 : 1 2 3 4 5 6 7 8 9 10 11 12 13
       Day 5 : 1 2 3 4 5 6 7 8 9 10 11 12 13
```

2.4 Επιλογή των βέλτιστων τιμών των παραμέτρων της εφαρμογής Βελτιστοποίησης Ωρολογίου Προγράμματος με Γενετικό Αλγόριθμο.

Για την βέλτιστη ρύθμιση των παραμέτρων του ΓΑ πραγματοποιήθηκε ένας σημαντικός αριθμός πειραμάτων, ώστε να καταλήξουμε στο ποιες θα είναι οι βέλτιστες παράμετροι λειτουργίας του ΓΑ (πληθυσμός, όριο γενεών, πιθανότητες εφαρμογής τελεστών, κ.λ.π.), και ποιοι είναι οι βέλτιστοι γενετικοί τελεστές που βοηθούν τον ΓΑ να προσεγγίσει κατά το δυνατόν τη βέλτιστη λύση.
Στα πειράματα τα οποία πραγματοποιήσαμε χρησιμοποιήσαμε ένα πλήθος παραμέτρων. Σε πολλές από αυτές χρησιμοποιήσαμε περισσότερες από μια εναλλακτικές τιμές. Οι παράμετροι οι οποίες χρησιμοποιήσαμε καθώς και οι διαφορετικές τιμές αυτών παραθέτονται στον Πίνακα 1 που ακολουθεί.

Κωδικός	**Κατηγορία παραμέτρων**	**Όνομα παραμέτρου**	Τιμή Παραμέτρου
1	Βασικές (Default) παράμετροι		
2	Standard Operators	Crossover	20 σημείων
3	Standard Operators	Crossover	Uniform crossover
4	Standard Operators	Mutation points	10 σημείων
5	Standard Operators	Mutation points	20 σημείων
6	Other Operators	Window Mutation	0.1 (πιθανότητα)
7	Other Operators	Window Mutation	0.3 (πιθανότητα)
8	Other Operators	Swap Chromosome	0.1 (πιθανότητα)
9	Other operator parameters	Swap Chromosome	0.3 (πιθανότητα)
10	Other operator parameters	Swap Bit	0.1 (πιθανότητα)
11	Other operator parameters	Swap Bit	0.3 (πιθανότητα)
12	Other operator parameters	Swap Window	0.1 (πιθανότητα)
13	Other operator parameters	Swap Window	0.3 (πιθανότητα)
14	Other operator parameters	Random Genotype	0.1 (πιθανότητα)
15	Other operator parameters	Random Genotype	0.3 (πιθανότητα)
16	Other operator parameters	Mutation Chromosome	0.1 (πιθανότητα)
17	Other operator parameters	Mutation Chromosome	0.3 (πιθανότητα)
18	Hill Climbing Operators	Bit Swap-Mutate	0.5 (πιθανότητα)
10	Hill Climbing Operators	Window Swap	0.5 (πιθανότητα)
20	Varying Fitness Function	Linear	5.000 γενεών
21	Varying Fitness Function	Square	5.000 γενεών
22	Varying Fitness Function	Exponential	5.000 γενεών
23	Standard Parameters	Generation Limit	10000 γενιές
24	Standard Parameters	Generation Limit	20000 γενιές
25	Standard Parameters	Population	100 (άτομα πληθυσμού)
26	Standard Parameters	Population	200 (άτομα πληθυσμού)

27	Advanced Hill Climbing	Micro GA Combinatorial	1.0 (πιθανότητα)
28	Advanced Hill Climbing	DayClimb + FixTeacher + FixRoom	1.0 (πιθανότητα)

Πίνακας 1: Παράμετροι που χρησιμοποιήθηκαν στην εφαρμογή Βελτιστοποίησης Ωρολογίου Προγράμματος με Γενετικό Αλγόριθμο.

Όπου:

1. Το **Uniform Crossover** είναι μία παραλλαγή του Crossover Πολλαπλών Σημείων, όπου κάθε bit του γονότυπου του απόγονου λαμβάνεται με ίση πιθανότητα από τον έναν ή τον άλλο γονέα.
2. Το **Window Mutation** είναι ένας ειδικός τελεστής ο οποίος επιλέγει τυχαία ένα χρωμόσωμα (κωδικοποιημένο sub-string μεταβλητής) μέσα στον γονότυπο, καθώς και ένα παράθυρο συνεχόμενων bits μέσα στο χρωμόσωμα και μεταλλάσσει το συνολικό παράθυρο των bits μετατρέποντας τα bits είτε όλα σε ένα είτε όλα σε μηδέν με ίση πιθανότητα.
3. Το **Swap Chromosome** είναι ένας ειδικός τελεστής ο οποίος επιλέγει τυχαία δυο χρωμοσώματα του γονότυπου (τα κωδικοποιημένα sub-strings των δυο μεταβλητών) και τα αντιμεταθέτει μέσα στον γονότυπο.
4. Το **Swap Bit** είναι ένας ειδικός τελεστής ο οποίος επιλέγει τυχαία δύο χρωμοσώματα (κωδικοποιημένα sub-strings μεταβλητών) του γονότυπου και μία θέση bit μέσα στα χρωμοσώματα και αντιμεταθέτει τα αντίστοιχα bits των δύο χρωμοσωμάτων.
5. Το **Swap Window** είναι ένας ειδικός τελεστής ο οποίος επιλέγει δυο χρωμοσώματα (τα κωδικοποιημένα sub-strings των δυο μεταβλητών) του ιδίου γονότυπου τυχαία και επίσης επιλέγει τυχαία ένα παράθυρο συνεχόμενων bits μέσα στα χρωμοσώματα και εναλλάσσει τα αντίστοιχα bits των δυο sub-strings τα οποία περιέχονται μέσα στο παράθυρο.
6. Το **Random Genotype** είναι ένας ειδικός τελεστής ο οποίος επαναπροσδιορίζει με τυχαίο τρόπο τον συνολικό γονότυπο θέτοντας όλα τα bits αυτού ίσα με 0 ή 1 με ίση πιθανότητα.
7. Το **Mutation Chromosome** είναι ένας ειδικός τελεστής ο οποίος επιλέγει τυχαία ένα χρωμόσωμα του γονότυπου (κωδικοποιημένο sub-string μεταβλητής) και το μεταβάλλει, προσθέτοντας αριθμητικά σε αυτό έναν τυχαία επιλεγμένο δυαδικό αριθμό ο οποίος λαμβάνει τιμές μέσα από ένα συγκεκριμένο εύρος.
8. Το **Bit Swap-Mutate (Hill Climbing Operator)** είναι ένας ειδικός τελεστής αναρρίχησης ο οποίος επιλέγει τυχαία δυο χρωμοσώματα του γονότυπου (τα κωδικοποιημένα sub-strings των μεταβλητών) και για κάθε μία θέση bit των χρωμοσωμάτων πραγματοποιεί τα ακόλουθα:
   a. Αντιμεταθέτει τα αντίστοιχα bits των δυο χρωμοσωμάτων ή
   b. Μεταλλάσσει τα αντίστοιχα bits των δυο χρωμοσωμάτων

   Η πιθανότητα του να πραγματοποιηθεί είτε το a είτε το b είναι 0.5. Όντας τελεστής αναρρίχησης, το Bit Swap-Mutate απορρίπτει όσες λύσεις έχουν χειρότερη ποιότητα από την αρχική, ενώ κάθε φορά που βρίσκει μία λύση καλύτερη από την αρχική την υιοθετεί.
9. Το **Window Swap (Hill Climbing Operator)** είναι ένας ειδικός τελεστής αναρρίχησης ο οποίος επιλέγει τυχαία δυο χρωμοσώματα του γονότυπου (τα κωδικοποιημένα sub-strings των μεταβλητών) και καθορίζει δυο παράθυρα ιδίου μήκους τα οποία είναι ευθυγραμμισμένα ως προς την αρχή των δυο χρωμοσωμάτων, δηλαδή ξεκινάνε και τα δυο από το πρώτο bit αυτών. Στη συνέχεια ο τελεστής αντιμεταθέτει τα bits των δυο χρωμοσωμάτων τα οποία περιέχονται στα αντίστοιχα παράθυρα και αξιολογεί τον γονότυπο. Αν αυτός επιδεικνύει καλύτερη ποιότητα τότε

φυλάσσεται και σταματά η αντιμετάθεση, αλλιώς ο γονότυπος αποκαθίσταται και η αντιμετάθεση συνεχίζεται μεταθέτοντας και τα δυο παράθυρα κατά ένα bit δεξιότερα. Η παραπάνω διαδικασία επαναλαμβάνεται μέχρις ότου βρεθεί μια καλύτερη λύση ή τα δυο παράθυρα φτάσουν στο τέλος των χρωμοσωμάτων.

10. Το **Varying Fitness Function** (Μεταβλητή Συνάρτηση Ποιότητας) είναι στην ουσία μία καινοτόμος τεχνική διαχείρισης των περιορισμών που τίθενται σε προβλήματα βελτιστοποίησης. Σύμφωνα με την τεχνική αυτή οι όροι ποινής, που προστίθενται στην Συνάρτηση Ποιότητας για την παραβίαση των περιορισμών, δεν είναι σταθεροί σε σχέση με τον χρόνο αλλά μεταβάλλονται με μονότονα αύξοντα τρόπο. Έτσι οι ποινές είναι χαλαρότερες στην αρχή του ΓΑ και γίνονται αυστηρότερες προς το τέλος του ΓΑ. Η τεχνική αυτή έχει δείξει ότι βοηθά τον ΓΑ στον εντοπισμό των περιοχών καλής ποιότητας του χώρου λύσεων, αλλά και στην ικανοποίηση των περιορισμών. Έτσι λοιπόν οι ποινές παραβίασης των περιορισμών πολλαπλασιάζονται επί ένα μεταβαλλόμενο συντελεστή ποινής (Penalty Factor) V(g) που εξαρτάται από την γενιά g.
    Ο ρυθμός αύξησης των ποινών μπορεί να είναι :
    a. **Γραμμικός (Linear)** οπότε Penalty Factor = $V(g) = g/G$
    b. **Τετραγωνικός (Square)** οπότε Penalty Factor = $V(g) = \left(g/G\right)^2$
    c. **Εκθετικός (Exponential)** οπότε Penalty Factor = $V(g) = 1 - e^{-5 \cdot g/G}$
    (όπου g είναι ένας μετρητής γενεών ξεκινώντας από την γενιά στην οποία βρίσκεται ο Γενετικός Αλγόριθμος όταν η τεχνική της Συνάρτηση Μεταβλητής Ποιότητας (Varying Fitness Function - VFF) ενεργοποιηθεί και G είναι ο συνολικός αριθμός των γενεών όπου η Συνάρτηση Μεταβλητής Ποιότητας είναι ενεργοποιημένη).
11. Το **Population** καθορίζει το πλήθος των ατόμων που απαρτίζουν τον εξελισσόμενο πληθυσμό λύσεων του γενετικού αλγορίθμου. Στην ουσία αποτελεί το αριθμό των παραγόμενων λύσεων σε κάθε γενιά.
12. Ο προηγμένος **Hill Climbing τελεστής MGAC** (Micro GA Combinatorial Hill Climbing Operator) είναι ένας τελεστής που ψάχνει στην «γειτονιά» της καλύτερης κάθε φορά λύσης, με τη βοήθεια ενός meta-level ΓΑ μικρής κλίμακας (Micro GA). Έτσι λαμβάνοντας σε κάθε γενιά του κυρίως ΓΑ την καλύτερη μέχρι στιγμής λύση, ξεκινά ένας δεύτερος ΓΑ που εκτελείται με πληθυσμό 5 ατόμων και για 7 μόλις γενιές, ο οποίος ψάχνει τον χώρο των «συνδυαστικών γειτονιών» με μέγεθος 4 bits, που υπάρχουν «γύρω» από την καλύτερη λύση. Αν στο τέλος του Micro GA βρεθεί ακόμα καλύτερη λύση τότε εμφυτεύεται μέσα στην λύση του κυρίως ΓΑ.
13. Οι προηγμένοι τελεστές DayClimb, FixTeacher και FixRoom λειτουργούν και πάλι πάνω στην καλύτερη λύση κάθε γενιάς του κυρίως ΓΑ. Ο τρόπος λειτουργίας τους είναι ο ακόλουθος:
    DayClimb: Προσδιορίζεται εάν στην τρέχουσα λύση υπάρχουν events που παραβιάζουν το όριο ωρών ανά ημέρα. Αν υπάρχουν τότε λαμβάνεται τυχαία ένα τέτοιο event και δοκιμάζεται η μεταφορά του σε όλες τις άλλες ημέρες του προγράμματος, μία προς μία. Αν κατά τις μεταφορές προκύψει καλύτερη λύση τότε αντικαθιστά την προηγούμενη.
    FixTeacher: Προσδιορίζεται εάν στην τρέχουσα λύση υπάρχουν events που παραβιάζουν τους περιορισμούς ανάθεσης καθηγητών σε μαθήματα. Αν υπάρχουν τότε λαμβάνεται τυχαία ένα τέτοιο event και δοκιμάζεται να ανατεθεί σε έναν από τους καθηγητές που μπορούν να διδάξουν το μάθημα. Η δοκιμή γίνεται για όλους

τους καθηγητές που μπορούν να διδάξουν το μάθημα έναν προς έναν. Αν κατά τις δοκιμές προκύψει καλύτερη λύση τότε αντικαθιστά την προηγούμενη.

FixRoom: Προσδιορίζεται εάν στην τρέχουσα λύση υπάρχουν events που παραβιάζουν τους περιορισμούς ανάθεσης μαθημάτων σε αίθουσες. Αν υπάρχουν τότε λαμβάνεται τυχαία ένα τέτοιο event και δοκιμάζεται να τοποθετηθεί σε μία από τις αίθουσες στις οποίες μπορεί να διεξαχθεί το μάθημα. Η δοκιμή γίνεται για όλες τις αίθουσες που μπορούν να φιλοξενήσουν το μάθημα μία προς μία. Αν κατά τις δοκιμές προκύψει καλύτερη λύση τότε αντικαθιστά την προηγούμενη.

2.5 Τα αποτελέσματα των πειραμάτων.

Τα πειράματα τα οποία πραγματοποιήσαμε υπάκουαν στην παρακάτω φιλοσοφία: δοθέντων των παραπάνω παραμέτρων πραγματοποιούμε τα πειράματα μας με τις προκαθορισμένες τιμές τις οποίες περιέχει το περιβάλλον της εφαρμογής Βελτιστοποίησης Ωρολογίου Προγράμματος με Γενετικό Αλγόριθμο και εξετάζουμε κάθε φορά μια από τις παραπάνω παραμέτρους. Δεδομένου ότι δεν είναι δυνατό να δοκιμαστούν όλοι οι δυνατοί συνδυασμοί των παραμέτρων λόγω του γεγονότος ότι το πλήθος τους είναι ένας αριθμός απαγορευτικά μεγάλος, υιοθετείται η διαδικασία ελέγχου μιας παραμέτρου κάθε φορά. Έτσι, κατά τον έλεγχο της κάθε τιμής παραμέτρου, αν διαπιστωθεί ότι η εν λόγω τιμή βελτιώνει την απόδοση του γενετικού αλγορίθμου τότε ενσωματώνεται στον ΓΑ και χρησιμοποιείται σε όλα τα περαιτέρω πειράματα. Στην αντίθετη περίπτωση η παράμετρος απορρίπτεται. Η μέθοδος αυτή βασίζεται στην σχετική ανεξαρτησία των παραμέτρων του ΓΑ, η οποία επιβεβαιώνεται και βάσει των πειραματικών αποτελεσμάτων. Με αυτόν τον τρόπο εξετάζονται όλες οι παραπάνω παράμετροι και επιλέγονται εκείνες οι οποίες οδήγησαν στην βελτίωση της απόδοσης του Γενετικού Αλγορίθμου.

Συνδυασμός Παραμέτρων	**Mean Quality**	**Best Quality**	**Worst Quality**
Βασικές (Default)	74192	61087	97073
STANDARD OPERATORS			
Βασικές + παράμετρος 2	72387	52087	91080
Βασικές + παράμετρος 3 (υιοθετήθηκε)	**67980**	**60102**	**76077**
Βασικές + παράμετροι 3, 4	72482	57079	83072
Βασικές + παράμετρος 3, 5	81798	65113	93120
OTHER OPERATORS			
Βασικές + παράμετροι 3, 6	61486	50092	76090
Βασικές + παράμετρος 3, 7 (υιοθετήθηκε)	**61596**	**44118**	**78093**
Βασικές + παράμετρος 3, 7, 8 (υιοθετήθηκε)	**66387**	**42095**	**83094**
Βασικές + παράμετροι 3, 7, 9	65190	52094	76094
Βασικές + παράμετροι 3, 7, 8, 10	70192	56091	83101
Βασικές + παράμετροι 3, 7, 8, 11	65991	53101	83085
Βασικές + παράμετροι 3, 7, 8, 12	65992	54092	88094
Βασικές + παράμετροι 3, 7, 8, 13	65288	56094	76072
Βασικές + παράμετροι 3, 7, 8, 14	71793	61117	80096
Βασικές + παράμετροι 3, 7, 8, 15	76979	52102	89087
Βασικές + παράμετροι 3, 7, 8, 16 (υιοθετήθηκε)	**53590**	**45093**	**65079**
Βασικές + παράμετροι 3, 7, 8, 17	59789	45097	77103
HILL CLIMBING OPERATORS			
Βασικές + παράμετροι 3, 7, 8, 16, 18	68581	46085	92080
Βασικές + παράμετροι 3, 7, 8, 16, 19	58389	48098	71101

Varying Fitness Function			
Βασικές + παράμετροι 3, 7, 8, 16, 20	50370	39078	67089
Βασικές + παράμετροι 3, 7, 8, 16, 21 (υιοθετήθηκε)	**52359**	**37093**	**63036**
Βασικές + παράμετροι 3, 7, 8, 16, 22	57796	47764	69626
STANDARD PARAMETERS			
Βασικές + παράμετροι 3, 7, 8, 16, 21, 23	56285	43074	68078
Βασικές + παράμετροι 3, 7, 8, 16, 21, 24	51188	34085	78083
Βασικές + παράμετροι 3, 7, 8, 16, 21, 25	53388	39084	88098
Βασικές + παράμετροι 3, 7, 8, 16, 21, 26 (υιοθετήθηκε)	**42089**	**26104**	**59082**
Advanced Hill Climbing			
Βασικές + παράμετροι 3,7,8,16,21,26,27 (υιοθετήθηκε)	**29882**	**23082**	**35078**
Βασικές + παράμετροι 3,7,8,16,21,26,27,28 (υιοθετήθ.)	**18286**	**12533**	**29047**

Πίνακας 2: Αποτελέσματα της Καλύτερης Λύσης, της Χειρότερης Λύσης και της Μέσης Τιμής της ποιότητας των λύσεων όλων των πειραμάτων που προέκυψαν με χρήση συνδυασμού παραμέτρων.

Κάθε δοκιμή για συγκεκριμένη τιμή παραμέτρου περιλάμβανε 10 διαφορετικές εκτελέσεις με διαφορετική κάθε φορά αρχικοποίηση της γεννήτριας τυχαίων αριθμών. Ο αριθμός των 10 εκτελέσεων επιβάλλεται από την στοχαστική φύση των ΓΑ, που απαιτούν την αξιολόγησή τους επί στατιστικών στοιχείων που εξάγονται από ένα αριθμό πειραμάτων και όχι από ένα και μοναδικό πείραμα. Από τις 10 εκτελέσεις του ΓΑ τα στατιστικά στοιχεία που πήραμε και εμφανίζονται στον Πίνακα 2, είναι : Η Μέση Τιμή Ποιότητας των 10 εκτελέσεων, Η καλύτερη Ποιότητα των 10 εκτελέσεων και η Χειρότερη Ποιότητα των 10 εκτελέσεων.

Οι Βασικές τιμές των παραμέτρων είναι οι ακόλουθες:

ΟΝΟΜΑ ΜΕΤΑΒΛΗΤΗΣ	**ΛΕΙΤΟΥΡΓΙΑ ΜΕΤΑΒΛΗΤΗΣ**
Bits Per Variable	συνολικός αριθμός events = 192, ενώ το πλήθος των bits που χρησιμοποιούνται για την «τοποθέτηση» κάθε μαθήματος είναι συνολικά 4992 δηλαδή αντιστοιχούν 26 bits ανά event.
Population:	= 50 που καθορίζει το πλήθος των γονότυπων που συνιστούν τον πληθυσμό του Γενετικού Αλγορίθμου.
Selection Method:	Roulette Wheel δηλαδή κάθε γονέας επιλέγεται με τη μέθοδο του Τροχού της Ρουλέτας με πιθανότητα ανάλογη της ποιότητάς του.
Multi Point Crossover	όπου κάθε γονέας χωρίζεται σε ένα αριθμό N+1 τμημάτων από N σημεία κοπής που επιλέγονται τυχαία και τα τμήματα ανταλλάσσονται
Crossover Points:	αριθμός σημείων που θα γίνει Crossover (ισούται με 5)
Mutation Points:	αριθμός σημείων που θα γίνει Mutation (ισούται με 5)
Adjustable Probabilities	οι πιθανότητες εφαρμογής των Mutation και Crossover προσαρμόζονται αυτόματα κατά τη διάρκεια της εκτέλεσης του ΓΑ.
Swap Bit Probability	πιθανότητα εφαρμογής του τελεστή Swap Bit = 0.0
Swap Window Probability	πιθανότητα εφαρμογής του τελεστή Swap Window = 0.0
Window Mutation Probability	πιθανότητα εφαρμογής του τελεστή Window Mutation = 0.0
Swap Chromosome Probability	πιθανότητα εφαρμογής του τελεστή Swap Chromosome = 0.0
Mute Chromosome Probability	πιθανότητα εφαρμογής του τελεστή Mutation

	Chromosome = 0.0
Rand Geno Probability:	πιθανότητα εφαρμογής του τελεστή Random Genotype = 0.0
BitSwapMute Probability	πιθανότητα εφαρμογής του τελεστή Bit Swap Mutation = 0.0
Win Swap Hill Climbing Probability	πιθανότητα εφαρμογής του τελεστή Window Swap Hill Climbing = 0.0
Varying Fitness Function	η μεταβλητή συνάρτηση ποιότητας στις βασικές παραμέτρους ΔΕΝ εφαρμόζεται.
Implement Elitism	εφαρμόζεται η τεχνική του Ελιτισμού
Generation Limit	όριο γενεών των γενετικών και ισούται με 5000
No Of Tasks	εκτελούνται δέκα πειράματα για κάθε τιμή παραμέτρου

Πίνακας 3: Βασικές παράμετροι που χρησιμοποιήθηκαν στην εφαρμογή Βελτιστοποίησης Ωρολογίου Προγράμματος με Γενετικό Αλγόριθμο.

2.6 Σχολιασμός των αποτελεσμάτων.

Ο Πίνακας 2 περιέχει τα αποτελέσματα των πειραμάτων τα οποία πραγματοποιήθηκαν. Για κάθε γραμμή του πίνακα έχουν πραγματοποιηθεί 10 πειράματα, ώστε να εξαχθούν στατιστικά στοιχεία απόδοσης, καθώς οι ΓΑ είναι στοχαστικές μέθοδοι βελτιστοποίησης. Η απόδοση του Γενετικού Αλγορίθμου σε κάθε σετ δέκα πειραμάτων καταγράφεται με τις τιμές της Καλύτερης Λύσης των 10 πειραμάτων, της Χειρότερης Λύσης των 10 πειραμάτων και της Μέσης Τιμής της ποιότητας των λύσεων των 10 πειραμάτων. Η ποιότητα κάθε λύσης προκύπτει ως άθροισμα της αντικειμενικής ποιότητας (Objective Fitness Value) που έχει να κάνει με τους στόχους βελτιστοποίησης (π.χ. ελαχιστοποίηση κενών ωρών κ.λ.π.) ΣΥΝ τις ποινές για παραβίαση περιορισμών (Penalty Value):
Fitness = Objective Fitness Value + Penalty Value (minimization problem),

Όπου το Penalty Value προκύπτει ως το γινόμενο των παραβιαζόμενων περιορισμών, N_{viol} επί ένα συντελεστή ποινής που εδώ είναι το 1000. Επομένως, Penalty Value = N_{viol} x 1000.

Όπως διαφαίνεται και από τον Πίνακα 2 αρχικά πραγματοποιήθηκαν πειράματα με τις Βασικές παραμέτρους. Στη συνέχεια ελέγχθηκε η απόδοση του ΓΑ με διαφοροποίηση του τελεστή Crossover. Έτσι δοκιμάστηκε το Crossover 20 σημείων και στη συνέχεια το Uniform Crossover (παράμετροι 2 και 3). Από αυτές καλύτερη απόδοση παρουσίασε το Uniform Crossover παράμετρος η οποία και υιοθετήθηκε για τα επόμενα πειράματα. Η ίδια μεθοδολογία ακολουθήθηκε και για τις επόμενες παραμέτρους που ελέγχθηκαν. Κατά την διεξαγωγή των πειραμάτων βρέθηκε ότι βελτιώνουν την απόδοση του ΓΑ και υιοθετήθηκαν πέραν των βασικών οι εξής παράμετροι :

1. Το Uniform Crossover (παράμετρος με αρ. 3)
2. Ο τελεστής Window Mutation με τιμή πιθανότητας εφαρμογής = 0.3 (παράμετρος με αρ. 7)
3. Ο τελεστής Swap Chromosome με τιμή πιθανότητας εφαρμογής = 0.1 (παράμετρος με αρ. 8)
4. Ο τελεστής Mutation Chromosome με τιμή πιθανότητας εφαρμογής = 0.1 (παράμετρος με αρ. 16).
5. Η Μεταβλητή Συνάρτηση Ποιότητας με ρυθμό αύξησης ποινών = Linear (Γραμμική) (παράμετρος με αρ. 21).
6. Ο πληθυσμός 200 ατόμων (παράμετρος με αρ. 26).
7. Ο τελεστής αναρρίχησης Micro GA Combinatorial (παράμετρος με αρ. 27).

8. Οι προηγμένοι τελεστές αναρρίχησης DayClimb, FixTeacher και FixRoom (παράμετρος με αρ. 28).

Ο συνδυασμός των παραπάνω παραμέτρων (πέραν των βασικών του Πίνακα 3) μας έδωσε τα καλύτερα αποτελέσματα κατά την διάρκεια των δοκιμών. Ιδιαίτερα πρέπει να σημειώσουμε την σημαντική βελτίωση που εισήγαγαν τόσο ο τελεστής αναρρίχησης Micro GA Combinatorial (παράμετρος με αρ. 27), όσο και οι ειδικοί προηγμένοι τελεστές αναρρίχησης DayClimb, FixTeacher και FixRoom (παράμετρος με αρ. 28).
Τα καλύτερα λοιπόν αποτελέσματα ήταν (για 10 πειράματα) :
Μέση Τιμή Ποιότητας Λύσεων : 18286
Καλύτερη Λύση : 12533
Χειρότερη Λύση : 29047

Η τιμή 12533 της καλύτερης λύσης όλων των πειραμάτων μπορεί να αναλυθεί ως εξής :
Fitness Value = 12533 = Objective Value + Penalty Value = 1038 + 11495
Η τιμή Penalty = 11495 σημαίνει ότι στην βέλτιστη λύση που παρήγαγε ο ΓΑ δεν ικανοποιούνται 12 περιορισμοί που έχουν τεθεί από το Setup File του προβλήματος. Οι περιορισμοί αυτοί όμως ανήκουν στη κατηγορία των Soft Constraints, δηλαδή είναι περιορισμοί μη κρίσιμοι για την λύση του προβλήματος. Τέτοιοι περιορισμοί είναι π.χ. η υπέρβαση των ορίων των διδακτικών ωρών/εβδομάδα ανά καθηγητή (ελάχιστο όριο , μέγιστο όριο), ή η υπέρβαση του ωραρίου της ημέρας κατά μία ώρα. Επίσης πρέπει να σημειωθεί ότι ο αριθμός των παραβιαζόμενων περιορισμών είναι εξαιρετικά μικρός καθώς αφορά όλο το ωρολόγιο πρόγραμμα του Τμήματος (7 εξάμηνα) και ότι ο αρχικός αριθμός παραβιαζόμενων περιορισμών σε μία λύση που παράγεται στα αρχικά στάδια του ΓΑ υπερβαίνει τους 600 παραβιαζόμενους περιορισμούς !
Η ποιότητα της λύσης ως προς την ικανοποίηση των στόχων (Objective Value) είχε τιμή 1038 που είναι καλύτερη από την αντίστοιχη ποιότητα του ωρολόγιου προγράμματος που αναπτύχθηκε από τους καθηγητές του ΤΕΙ Σερρών η οποία από την ίδια συνάρτηση ποιότητας αξιολογήθηκε με :
Objective Value = 1294
Penalty Value = 0.0
Πρέπει να σημειωθεί εδώ ότι οι λύσεις του ΓΑ θα ήταν σαφώς περισσότερο κοντά στο βέλτιστο (penalty=0) εάν αφήναμε τον ΓΑ να εκτελεστεί για μεγαλύτερο περιθώριο γενεών, το οποίο απλά δεν έγινε στη διάρκεια των δοκιμών, αλλά σαφώς και μπορεί να γίνει στην πράξη.
Επομένως μπορούμε συμπερασματικά να πούμε ότι η λύση του ΓΑ υπολείπεται ελαφρά σε σύγκριση με την λύση του ανθρώπου ως προς την ικανοποίηση των περιορισμών, αλλά υπερέχει ως προς την ικανοποίηση των στόχων του προβλήματος. Το γεγονός αυτό σε συνδυασμό με το ότι μέσω του ΓΑ η παραγωγή του ωρολόγιου προγράμματος γίνεται αυτόματα και σε μικρό χρονικό διάστημα, ενώ μπορεί να επανεκτελεστεί όσες φορές το επιθυμεί ο χρήστης ώστε να επαναληφθεί η αναζήτηση, καταδεικνύουν την ανωτερότητα της αναπτυχθείσας μεθόδου για τον «Αυτόματο Υπολογισμό Ωρολόγιων Προγραμμάτων με βάση τους Γενετικούς Αλγορίθμους».

Δρ. Σπυρίδων Α. Καζαρλής

Επίκουρος Καθηγητής
Επιστημονικός Υπεύθυνος