



**ΕΘΝΙΚΟΝ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟΝ
ΠΑΝΕΠΙΣΤΗΜΙΟΝ ΑΘΗΝΩΝ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ**

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ & ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΑΠΟΤΙΜΗΣΗ ΤΟΥ ΕΡΓΟΥ ΤΟΥ ΤΜΗΜΑΤΟΣ

**ΣΤΟΧΟΙ
ΓΙΑ ΤΗΝ ΠΕΡΙΟΔΟ 2008 - 2013**

ΠΑΡΑΡΤΗΜΑ ΙΧ

**ΕΠΙΛΟΓΗ ΠΤΥΧΙΑΚΩΝ & ΔΙΠΛΩΜΑΤΙΚΩΝ ΕΡΓΑΣΙΩΝ
ΤΟΜΟΣ 2008**



ΜΑΡΤΙΟΣ 2009

ΠΑΝΕΠΙΣΤΗΜΙΟΥΠΟΛΗ - ΑΘΗΝΑ 15784

Τηλ.: 210 727 5161 , FAX: 210 727 5214 , e-mail: secret@di.uoa.gr



ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

**ΕΠΙΛΕΓΜΕΝΕΣ
ΠΤΥΧΙΑΚΕΣ ΚΑΙ ΔΙΠΛΩΜΑΤΙΚΕΣ ΕΡΓΑΣΙΕΣ**

ΤΟΜΟΣ 5



ΑΘΗΝΑ 2008

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ
ΕΠΙΛΕΓΜΕΝΕΣ ΠΤΥΧΙΑΚΕΣ ΚΑΙ ΔΙΠΛΩΜΑΤΙΚΕΣ ΕΡΓΑΣΙΕΣ
ΤΟΜΟΣ 5

Εκδίδεται μία φορά το χρόνο από το:
Τμήμα Πληροφορικής και Τηλεπικοινωνιών
του Εθνικού και Καποδιστριακού Πανεπιστημίου Αθηνών,
Πανεπιστημιούπολη, 15784 Αθήνα
Τηλ: 2107275190, Fax:2107275333
Email: library@di.uoa.gr
URL: www.di.uoa.gr/lib

Επιμέλεια έκδοσης:
Επιτροπή Ερευνητικών και Αναπτυξιακών Δραστηριοτήτων

Ι.Ζ. Εμίρης (πρόεδρος),
Δ. Γουνόπουλος, Η. Μανωλάκος, Π. Ροντογιάννης

ISSN: 1790-4544

Copyright © 2008
Τμήμα Πληροφορικής και Τηλεπικοινωνιών,
Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών

ΠΡΟΛΟΓΟΣ

Ο τόμος αυτός περιλαμβάνει περιλήψεις των καλύτερων διπλωματικών και πτυχιακών εργασιών που εκπονήθηκαν στο Τμήμα Πληροφορικής και Τηλεπικοινωνιών του Εθνικού και Καποδιστριακού Πανεπιστημίου Αθηνών στο διάστημα 1/1/2007 - 31/12/2007. Πρόκειται για τον 5^ο τόμο στη σειρά αυτή. Στόχος του θεσμού είναι η ενθάρρυνση της δημιουργικής προσπάθειας και η προβολή των πρωτότυπων εργασιών των φοιτητών του Τμήματος.

Η συμμετοχή των φοιτητών ήταν και φέτος πολύ ικανοποιητική. Κατόπιν της διαδικασίας κρίσεως, έγιναν τελικά δεκτές για δημοσίευση 13 συνολικά διπλωματικές και πτυχιακές εργασίες. Η στάθμη των εργασιών που υποβλήθηκαν ήταν υψηλή και τα γνωστικά αντικείμενα που πραγματεύονται οι εργασίες καλύπτουν σε μεγάλο βαθμό ολόκληρη την περιοχή της Πληροφορικής και των Τηλεπικοινωνιών.

Κλείνοντας τον πρόλογο αυτό, θα θέλαμε να ευχαριστήσουμε τους φοιτητές για τον χρόνο που αφιέρωσαν για να παρουσιάσουν τη δουλειά τους στα πλαίσια αυτού του θεσμού. Ελπίζουμε η διαδικασία αυτή να προσέφερε και στους ίδιους μια εμπειρία που θα τους βοηθήσει στη συνέχεια των σπουδών τους ή της επαγγελματικής τους σταδιοδρομίας. Θα επιθυμούσαμε να ευχαριστήσουμε ιδιαίτερα τους συναδέλφους, οι οποίοι δέχθηκαν να είναι κριτές, για τον χρόνο που αφιέρωσαν. Τέλος, ευχαριστούμε την κ. Ιακωβίνα Κανάκη για την υποστήριξη της παρούσας έκδοσης. Ο πίνακας του εξωφύλλου ανήκει στον Κωνσταντίνο Βολανάκη.

Η Επιτροπή Ερευνητικών και Αναπτυξιακών Δραστηριοτήτων

Ο Πρόεδρος:

I.Z. Εμίρης

Τα Μέλη:

Δ. Γουνόπουλος

Π. Ροντογιάννης

Η. Μανωλάκος

Αθήνα, Σεπτέμβριος 2008

ΠΕΡΙΕΧΟΜΕΝΑ

| | |
|---|------------|
| Περιεχόμενα | i |
| Διπλωματικές Εργασίες | 1 |
| Άγγελος Αλεξόπουλος (επιβλέποντες: Μαρία Γρηγοριάδου και Ευαγγελία Γουλή) | 3 |
| Ιωάννης Ανδρουλακακης (επιβλέπων: Δημήτριος Συβρίδης) | 13 |
| Δημήτρης Διώχνος (επιβλέπων: Ιωάννης Εμίρης) | 23 |
| Φοίβος Θεοχάρης (επιβλέπων: Σταματόπουλος Παναγιώτης) | 33 |
| Ευστράτιος Κεραμιδάς (επιβλέπων: Δημήτρης Μαρούλης) | 43 |
| Ευάγγελος Λογαράς (επιβλέπων: Η. Μανωλάκος) | 53 |
| Δημήτριος Μανατάκης (επιβλέπων: Η. Μανωλάκος) | 65 |
| Διονύσιος Ξενάκης , Δημήτριος Τσόλκας (επιβλέπων: Λάζαρος Μεράκος) | 75 |
| Εύα Σιταρίδη (επιβλέπων: Γιάννης Ιωαννίδης, Φραγκίσκος Πεντάρης) | 89 |
| Αναστάσιος Σπηλιωτόπουλος (επιβλέπων: Δρακούλης Μαρτάκος) | 95 |
| Πτυχιακές Εργασίες | 105 |
| Ιωάννης Κορμαρής (επιβλέπων: Αφροδίτη Τσαλγατίδου) | 107 |
| Παύλος Σουλδάτος, Σταύρος Πολίτης (επιβλέπων: Σταματόπουλος Παναγιώτης) | 119 |
| Ιωάννης Σταμούλιας, Κωνσταντίνος Μασούνης (επιβλέπων: Πασχάλης Αντώνιος) | 129 |

Διπλωματικές Εργασίες

Ανάπτυξη Συγγραφικού Εργαλείου Δραστηριοτήτων Εννοιολογικής Χαρτογράφησης

Άγγελος Α. Αλεξόπουλος

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών,
Τμήμα Πληροφορικής και Τηλεπικοινωνιών

Περίληψη: Στα πλαίσια της παρούσας διπλωματικής εργασίας, αναπτύχθηκε ένα εργαλείο συγγραφής εκπαιδευτικών δραστηριοτήτων εννοιολογικής χαρτογράφησης. Οι εννοιολογικοί χάρτες αποτελούν ένα γραφικό τρόπο οργάνωσης και αναπαράστασης της εννοιολογικής δομής των μαθητών. Παράλληλα, έχει χρησιμοποιηθεί ως μια διδακτική τεχνική και εκπαιδευτική στρατηγική που έχει σκοπό να ενισχύσει την εποικοδομητική μάθηση και τη μάθηση με νόημα. Ο στόχος του εργαλείου που αναπτύχθηκε είναι τριπλός. Αρχικά, ο εκπαιδευτικός μπορεί να δημιουργήσει δραστηριότητες πάνω σε γνωστικά αντικείμενα τις οποίες οι εκπαιδευόμενοι καλούνται να εκπονήσουν μέσα από το μαθησιακό περιβάλλον εννοιολογικής χαρτογράφησης COMPASS. Επίσης, μέσα από το συγκεκριμένο εργαλείο ο εκπαιδευτικός μπορεί να παρακολουθήσει τη μαθησιακή πορεία των εκπαιδευόμενων και τέλος οι συγγραφείς δραστηριοτήτων εννοιολογικής χαρτογράφησης μπορεί να συνεργάζονται και να αξιολογούν τις δραστηριότητες που κατασκευάζουν.

Λέξεις κλειδιά: έννοια, σύνδεσμος, εκπαιδευτική δραστηριότητα, εκπαιδευτικό εργαλείο, εννοιολογική χαρτογράφηση

1 Εισαγωγή¹

Οι Εννοιολογικοί Χάρτες (Concept Maps) είναι ένα πολύτιμο εργαλείο διδασκαλίας και αξιολόγησης της εκπαιδευτικής διαδικασίας αφού ενισχύουν την μάθηση, προωθούν την δημιουργικότητα και επιτρέπουν στους εκπαιδευόμενους να εξωτερικεύσουν τις γνωσιολογικές δομές που έχουν ήδη αποκτηθεί [9]. Σίγουρα ο πιο φυσικός και απλός τρόπος κατασκευής εννοιολογικών χαρτών φαίνεται να είναι η χρήση χαρτιού και μολυβιού ή ακόμα και η χρήση σημειωμάτων “post - it”. Παρ’ όλα αυτά, ο συγκεκριμένος τρόπος κατασκευής εννοιολογικών χαρτών εκτός του ότι επιβάλλει κάποιους περιορισμούς, δυσκολεύει πολύ την αξιολόγηση, αναθεώρηση και ανατροφοδότηση ως εκπαιδευτικές διεργασίες. Με την χρήση της πληροφορικής είναι πλέον δυνατόν να αντισταθμιστούν όλες οι παραπάνω δυσκολίες αλλά και να

¹ Επιβλέποντες καθηγητές: Μαρία Γρηγοριάδου και Ευαγγελία Γουλή

προσθεθούν επιπλέον δυνατότητες στην κλασσική παιδαγωγική προσέγγιση των εννοιολογικών χαρτών.

Ο τομέας της Εννοιολογικής Χαρτογράφησης γνωρίζει μεγάλη άνθηση και έχουν αναπτυχθεί τόσο εμπορικά προγράμματα όπως το Inspiration² όσο και ερευνητικά προγράμματα. Στο ερευνητικό επίπεδο τα λογισμικά έχουν σαν στόχο: (i) την υποστήριξη των εκπαιδευόμενων κατά την κατασκευή εννοιολογικών χαρτών καθώς και την υποστήριξη των εκπαιδευτών στον σχεδιασμό και οργάνωση κατάλληλων μαθημάτων όπως το TPL-KATS [7], CM-ED [10] and LEO [3], [4] (ii) την προαγωγή και διευκόλυνση της συνεργατικής μάθησης όπως το Cmap Tools [1] (iii) την στήριξη αξιολόγησης κατά την εκπόνηση μιας δραστηριότητας όπως το Reasonable Fallible Analyser [5], Java Mapper [8] and Verified Concept Mapper [2].

Πάνω σε αυτή την γραμμή έρευνας και με στόχο την υποστήριξη όλων των διαδικασιών μάθησης αναπτύχθηκε ένα προσαρμοστικό περιβάλλον κατασκευής εννοιολογικών χαρτών που ονομάζεται COMPASS (Concept Map Assessment Tool). Το μαθησιακό περιβάλλον COMPASS³ έχει ως στόχο να υποστηρίξει τη διαδικασία της αξιολόγησης και τη μαθησιακή διαδικασία μέσα από την εκπόνηση δραστηριοτήτων εννοιολογικής χαρτογράφησης [6]. Το πεδίο γνώσης του COMPASS δεν αφορά ένα συγκεκριμένο γνωστικό αντικείμενο, αλλά μπορεί να χρησιμοποιηθεί σε οποιοδήποτε γνωστικό αντικείμενο κρίνεται ότι είναι εφαρμόσιμη η τεχνική της εννοιολογικής χαρτογράφησης. Συγκεκριμένα, το COMPASS: (a) επιτρέπει στους εκπαιδευόμενους να εκπονήσουν δραστηριότητες εννοιολογικής χαρτογράφησης, (b) αναλύει τους εννοιολογικούς χάρτες των εκπαιδευομένων προκειμένου να αναγνωρίσει τυχόν εσφαλμένες αντιλήψεις των εκπαιδευομένων και να διαγνώσει ποιοτικά και ποσοτικά το επίπεδο γνώσης τους, (c) αξιολογεί τους εννοιολογικούς χάρτες που δημιουργούνται από τους εκπαιδευόμενους βάση κριτηρίων αξιολόγησης που ορίζονται από τον εκπαιδευτικό και τέλος (d) καθοδηγεί, βοηθά και διδάσκει τους εκπαιδευόμενους παρέχοντάς διαφορετικούς τύπους και πολλαπλές μονάδες ανατροφοδότησης, προσαρμοσμένης στα ιδιαίτερα χαρακτηριστικά και στις ανάγκες των εκπαιδευομένων.

2 Το εργαλείο Compass Authoring Tool

Με στόχο την υποστήριξη των εκπαιδευτικών στην δημιουργία δραστηριοτήτων εννοιολογικής χαρτογράφησης και την παρακολούθηση την προόδου των μαθητών δημιουργήθηκε το εργαλείο CAT (Compass Authoring Tool) σαν βασικό κομμάτι του περιβάλλοντος Compass. Συγκεκριμένα το περιβάλλον CAT δίνει την δυνατότητα στους εκπαιδευτικούς να δημιουργήσουν δραστηριότητες εννοιολογικής χαρτογράφησης τις οποίες οι μαθητές εκπονούν μέσα από το περιβάλλον του Compass. Στην εικόνα 1 φαίνεται η αλληλεπίδραση μεταξύ του καθηγητή και μαθητή μέσω της εφαρμογής Compass και CAT.

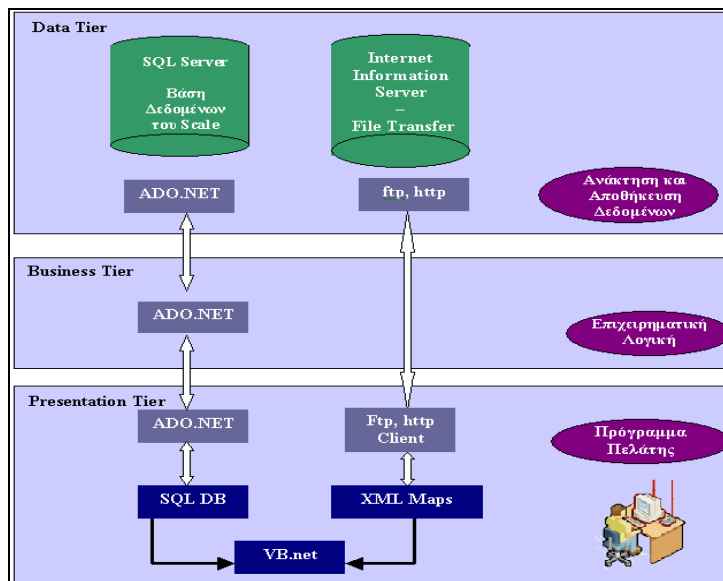
² <http://www.inspiration.com>

³ <http://hermes.di.uoa.gr/compass>

(e) Βοηθάει την συνεργασία των καθηγητών στην συγγραφή δραστηριοτήτων εννοιολογικής χαρτογράφησης. Συγκεκριμένα, οι εκπαιδευτικοί έχουν την δυνατότητα να έχουν πρόσβαση στην δραστηριότητες εννοιολογικής χαρτογράφησης που έχουν δημιουργήσει και δημοσιεύσει άλλοι εκπαιδευτικοί, να αξιολογήσουν και να σχολιάσουν δραστηριότητες που έχουν γράψει άλλοι συγγραφείς καθώς και να συνεργαστούν με άλλους συγγραφείς στην δημιουργία και αξιολόγηση υλικού ανατροφοδότησης για συγκεκριμένες δραστηριότητες μέσω ασύγχρονης επικοινωνίας.

2.1 Αρχιτεκτονική Περιβάλλοντος CAT

Το εργαλείο του εκπαιδευτή περιλαμβάνει ένα σύνολο από λειτουργικές μονάδες (modules) για την υλοποίηση των διαφόρων λειτουργιών του. Η υλοποίηση βασίζεται στην αρχιτεκτονική τριών επιπέδων (3 – tier architecture), όπως απεικονίζεται στην Εικόνα 2.



Εικόνα 2: Παρουσίαση αρχιτεκτονικής 3 επιπέδων του εργαλείου.

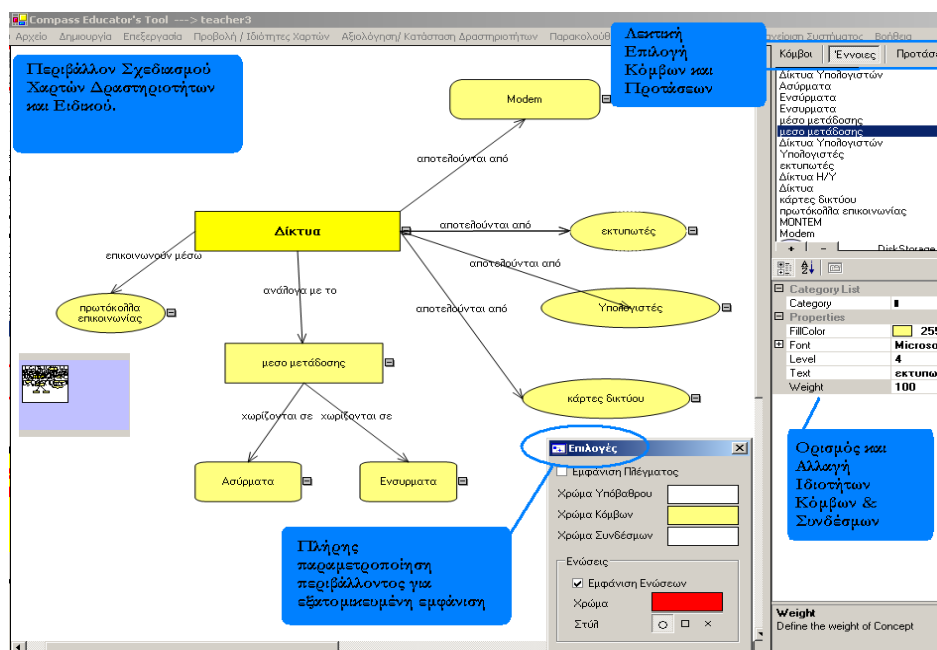
Συγκεκριμένα για το συγγραφικό εργαλείο τα 3 επίπεδα προσαρμόζονται στα εξής:

Data Tier – Υπάρχει μια Βάση Δεδομένων στο κεντρικό εξυπηρετητή η οποία φυλάει τα δεδομένα της εφαρμογής του Compass. Η Βάση Δεδομένων τρέχει σε Microsoft SQL Server. Επιπλέον, υπάρχει ένας ορισμένος κατάλογος στον κεντρικό εξυπηρετητή στον οποίο φυλάσσονται τα απαραίτητα αρχεία για την λειτουργία του Compass. Για την ακρίβεια υπάρχουν οι εννοιολογικοί χάρτες των χρηστών, οι εννοιολογικοί χάρτες των ειδικών, αρχεία καταγραφής της αλληλεπίδρασης των χρηστών με το Compass, επιπλέον συνοδευτικό εκπαιδευτικό υλικό καθώς και υλικό

ανατροφοδότησης για τις δραστηριότητες. Τα αρχεία αυτά είναι σε μορφή XML. Το επίπεδο των δεδομένων είναι ο συνδεδεμένος κρίκος του μαθησιακού περιβάλλοντος του Compass με το συγγραφικό εργαλείο που αναπτύχθηκε στο πλαίσιο της εργασίας.

Business Tier – Η ‘λογική’ των εφαρμογών. Από το στρώμα των δεδομένων ανακτώνται δεδομένα τα οποία μπορούν να είναι από διαφορετικούς πίνακες και όψεις της Βάσης Δεδομένων. Τα δεδομένα ανακτώνται μέσω του ADO.NET στα οποία εφαρμόζονται κάποιοι λογικοί μετασχηματισμοί για να τροφοδοτήσουν το στρώμα παρουσίασης της εφαρμογής. Επίσης, ισχύει και το αντίστροφο. Τα δεδομένα από τον χρήστη μετασχηματίζονται κατάλληλα για να αποθηκευτούν σωστά στην Βάση Δεδομένων. Το ADO.NET υποστηρίζει την αντικειμενοστραφή ανάπτυξη λογισμικού και γι αυτό στην συγκεκριμένη εφαρμογή όλοι οι μετασχηματισμοί γίνονται πάνω σε αντικείμενα τα οποία αντανακλούν τους πίνακες της Βάσης Δεδομένων.

Presentation Tier – Η γραφική διεπαφή (user interface) ανήκει στους rich clients. Πρόκειται για παραθυρικό περιβάλλον ανεπτυγμένο μόνο για Microsoft Windows με την βοήθεια του περιβάλλοντος προγραμματισμού VB.net. Από την Βάση Δεδομένων αντλούνται στοιχεία για τις δραστηριότητες του χρήστη ενώ από τον κατάλογο του κεντρικού εξυπηρετητή αντλούνται αρχεία χαρτών και ανατροφοδότησης. Το γραφικό περιβάλλον της εφαρμογής παρουσιάζεται στην εικόνα 3.



Εικόνα 3: Το γραφικό περιβάλλον εργασίας του CAT. Μια δραστηριότητα Εννοιολογικής Χαρτογράφησης με κεντρική έννοια τα "Δίκτυα" βρίσκεται σε εξέλιξη.

2.2 Κατανομή λειτουργιών σε ρόλους χρηστών

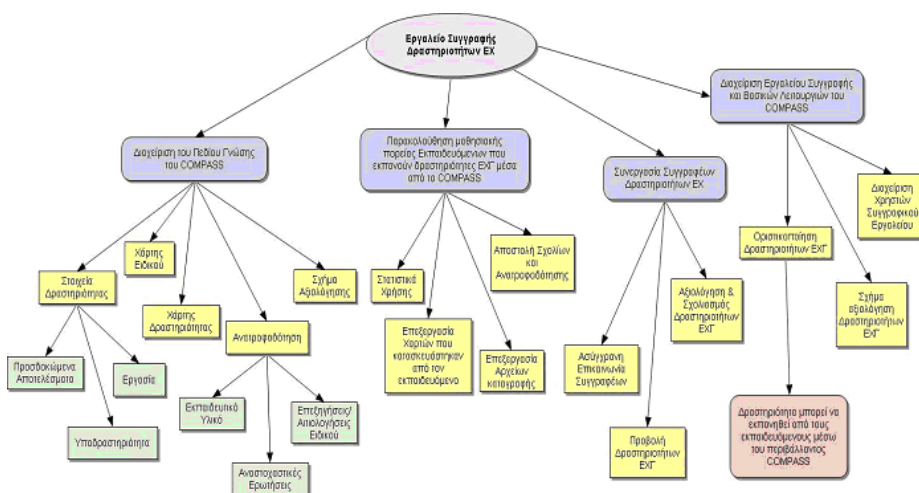
Για να επιτευχθούν οι παραπάνω λειτουργίες υπάρχουν σαφή διαχωριστικά όρια στους ρόλους των χρηστών του συστήματος. Συγκεκριμένα υπάρχουν οι παρακάτω τρεις διακριτοί ρόλοι:

Ο ρόλος του «**Συγγραφέα**». Ο Συγγραφέας μπορεί να σχεδιάσει και να αναπτύξει οποιαδήποτε δραστηριότητα. Επίσης, του δίνεται η δυνατότητα να προβάλλει και να αξιολογήσει δραστηριότητες εννοιολογικής χαρτογράφησης που έχουν γίνει από άλλους «Συγγραφείς» ή και «Εκπαιδευτές»

Ο ρόλος του «**Εκπαιδευτή**». Ο Εκπαιδευτής έχει περισσότερες αρμοδιότητες από τον Συγγραφέα και μπορεί επιπλέον να οριστικοποιήσει δραστηριότητες έτσι ώστε να μην μπορούν να γίνουν περαιτέρω αλλαγές από άλλους «Συγγραφείς» και «Εκπαιδευτές» καθώς και να παρακολουθήσει την μαθησιακή πορεία των εκπαιδευόμενων.

Ο ρόλος του «**Διαχειριστή**». Τέλος ο Διαχειριστής έχει τις περισσότερες αρμοδιότητες από όλους. Εκτός από τις δυνατότητες που έχουν ο Συγγραφέας και ο Εκπαιδευτής μπορεί επιπρόσθετα να αλλάξει τα δικαιώματα πρόσβασης όλων των χρηστών, αλλά και να καθορίσει εξ ολοκλήρου το γενικό πλαίσιο αξιολόγησης των δραστηριοτήτων.

Ο ρόλος κάθε χρήστη που μπαίνει στο σύστημα καθορίζεται κατά την είσοδο του στο σύστημα. Οι χρήστες πληκτρολογούν το όνομα χρήστη και τον κωδικό που τους έχει δοθεί και ανάλογα με τον ρόλο τους υπάρχουν ενεργές ή όχι οι επιλογές του προγράμματος. Μια συνοπτική διαγραμματική παρουσίαση των λειτουργιών του προγράμματος φαίνονται στην εικόνα 4.



Εικόνα 4: Συνοπτική Διαγραμματική Περιγραφή Λειτουργιών του Εργαλείου Συγγραφής Δραστηριοτήτων Εννοιολογικής Χαρτογράφησης

2.2.1 Διαχείριση του πεδίου γνώσης

Στην παρούσα ενότητα παρουσιάζεται βηματικά μια τυπική σειρά εκτέλεσης ενεργειών – λειτουργιών για την δημιουργία μιας δραστηριότητας από τον συγγραφέα. Τα βήματα τυπικά και δεν είναι υποχρεωτικό να ακολουθηθούν με την προκαθορισμένη σειρά.

Το πρώτο βήμα για την δημιουργία μιας δραστηριότητας εννοιολογικής χαρτογράφησης είναι ο ορισμός των βασικών χαρακτηριστικών της δραστηριότητας καθώς και ο ορισμός των λεπτομερειών της αντίστοιχης υποδραστηριότητας. Σε αυτό το βήμα ορίζεται το γνωστικό αντικείμενο της δραστηριότητας, η ερώτηση στόχος της δραστηριότητας, τα προσδοκώμενα αποτελέσματα καθώς και κάποιες παράμετροι για το είδος του εννοιολογικού χάρτη (Κατασκευή, Επέκταση, Διόρθωση,...), τον τρόπο με τον οποίο αλληλεπιδρά ο μαθητής με αυτόν (ελεύθερη εισαγωγή εννοιών – συνδέσμων ή επιλογή από λίστα) και την μορφή ανατροφοδότησης. Σε αυτό το σημείο ο συγγραφέας μπορεί να στείλει και κάποιο συνοδευτικό αρχείο (για παράδειγμα μια εικόνα) η οποία θα αποσαφηνίζει περισσότερο κάποια έννοια.

Το επόμενο βήμα, είναι η αλλαγή του σχήματος αξιολόγησης για την συγκεκριμένη δραστηριότητα. Ο συγγραφέας ανάλογα με τον στόχο που έχει θέσει για την δραστηριότητα έχει την δυνατότητα να προσαρμόσει το σχήμα αξιολόγησης στην συγκεκριμένη δραστηριότητα.

Η δημιουργία του χάρτη του ειδικού είναι το επόμενο σημαντικό βήμα για την ολοκλήρωση της δραστηριότητας. Το γραφικό περιβάλλον έχει γίνει αρκετά εύχρηστο και προσφέρει αρκετές επιλογές για το είδος του κόμβου και του συνδέσμου. Στο γραφικό περιβάλλον γίνονται όλες οι επιλογές με το ποντίκι. Ο συγγραφέας για κάθε έννοια, σύνδεσμο ή πρόταση έχει τον απόλυτο έλεγχο στις ιδιότητές τους. Ο τελικός χάρτης αποθηκεύεται στον κεντρικό εξυπηρετητή. Παράλληλα, όμως δίνεται η δυνατότητα προσωρινής αποθήκευσης τοπικά έτσι ώστε να συνεχιστεί η εργασία από εκεί που σταμάτησε. Τέλος, υπάρχουν οι επιλογές της εκτύπωσης και της δημοσίευσης του χάρτη σαν web σελίδα.

Αν το απαιτεί η εκπαιδευτική δραστηριότητα ο συγγραφέας μπορεί να δημιουργήσει και κάποιο χάρτη ο οποίος θα δοθεί στους εκπαιδευόμενους ανάλογα με τους στόχους της δραστηριότητας. Εδώ όμως, έχει την δυνατότητα να προστεθούν και έννοιες ή σύνδεσμοι οι οποίοι δεν θα εμφανίζονται στον χάρτη. Με αυτό τον τρόπο μπορεί να διαπιστωθεί το κατά πόσο ξεχωρίζουν οι μαθητές τις περιττές έννοιες από τις χρήσιμες.

Επιπλέον, υπάρχει η δυνατότητα να δημιουργηθεί εκπαιδευτικό υλικό τόσο σε επίπεδο γνωστικής έννοιας όσο και σε επίπεδο δραστηριότητας. Για την δημιουργία προτείνονται οι έννοιες που έχουν εισαχθεί στον χάρτη του ειδικού και ο χρήστης έχει την δυνατότητα να ορίσει εκπαιδευτικό υλικό (εικόνες, κείμενο, ήχος) , ερωτήσεις ανατροφοδότησης και επεξηγήσεις πάνω σε αυτές.

Αφού ολοκληρωθούν τα προηγούμενα στάδια, ο συγγραφέας μπορεί να κάνει την δραστηριότητα που έχει δημιουργήσει διαθέσιμη για αξιολόγηση και σχολιασμό από άλλους συγγραφείς έτσι ώστε να ανατροφοδοτηθεί η όλη διαδικασία της δημιουργίας της δραστηριότητας μέσω της συνεργασίας μεταξύ διαφορετικών εκπαιδευτικών.

Σε οποιοδήποτε σημείο της παραπάνω διαδικασίας, ο συγγραφέας έχει την δυνατότητα προβολής, επεξεργασίας και διαγραφής οποιουδήποτε βήματος. Το

τελικό βήμα είναι η οριστικοποίηση της δραστηριότητας έτσι ώστε να γίνει διαθέσιμη στους χρήστες του Compass.

2.2.2 Παρακολούθηση Μαθησιακής Πορείας

Οι εκπαιδευτές καθώς και οι διαχειριστές έχουν την δυνατότητα να παρακολουθούν την πορεία των εκπαιδευόμενων κατά την χρήση του Compass ως εργαλείου εκμάθησης. Έτσι αποκτούν μια ευρύτερη εικόνα για τις ιδιαίτερες ανάγκες του κάθε μαθητή αλλά και αξιολογούν τις δραστηριότητες με στόχο την βελτίωση αυτών ώστε να επικεντρώνονται στα σημεία που υπάρχει μεγαλύτερο πρόβλημα εκ μέρους των εκπαιδευόμενων. Τα στοιχεία αυτά φυλάσσονται από το Compass στην Βάση Δεδομένων του εξυπηρετητή.

Ειδικότερα, προσφέρονται οι παρακάτω δυνατότητες:

Εμφάνιση στατιστικών στοιχείων που αφορούν την εκπόνηση μιας δραστηριότητας εννοιολογικής χαρτογράφησης από τους μαθητές. Τα στατιστικά στοιχεία είναι ο συνολικός χρόνος εκπόνησης μιας δραστηριότητας, ο αριθμός φορών εκπόνησης, ο αριθμός φορών ανάλυσης του χάρτη, το είδος ανατροφοδότησης που επέλεξε ο μαθητής, το γνωστικό επίπεδο του μαθητή όπως αυτό εκτιμήθηκε από το σύστημα αυτόματα.

Ανάκτηση της τελευταίας καθώς και οποιασδήποτε προηγούμενης έκδοσης χάρτη του εκπαιδευόμενου.

Ανάκτηση των αρχείων καταγραφής, όπως αυτά δημιουργούνται κατά την διάρκεια εκπόνησης μιας δραστηριότητας. Τα αρχεία καταγραφής αποθηκεύουν όλη την αλληλεπίδραση του μαθητή με το σύστημα και έτσι ο εκπαιδευτής μπορεί να αποκτήσει μια πολύ ξεκάθαρη και λεπτομερή εικόνα για την πορεία του μαθητή.

Δυνατότητα αποστολής μέσω ηλεκτρονικού ταχυδρομείου σχολίων και ανατροφοδότησης από τον εκπαιδευτή προς τον μαθητή για τον τρόπο που αυτός εργάστηκε.

2.2.3 Ορισμός Σχήματος Αξιολόγησης Δραστηριοτήτων

Ο διαχειριστής του συστήματος έχει την δυνατότητα να μεταβάλλει το γενικό σχήμα αξιολόγησης των δραστηριοτήτων εννοιολογικής χαρτογράφησης. Για όλες τις δραστηριότητες υπάρχει ένα γενικό σχήμα αξιολόγησης με το οποίο το Compass βαθμολογεί αυτόματα και ποσοτικοποιεί το επίπεδο κατανόησης της κεντρικής έννοιας από τον εκπαιδευόμενο.

Ειδικότερα η αξιολόγηση των μαθητών έχει κατηγοριοποιηθεί στις παρακάτω κατηγορίες:

Μη πλήρης Σύνδεση
Μη ορθή Σύνδεση
Περιττή Σύνδεση
Μη αξιολογήσιμη Σύνδεση
Δεν υπάρχει πρόταση

Μη πλήρης Έννοια
Μη ορθή Έννοια
Περιττή Έννοια
Μη αξιολογήσιμη Έννοια

Για να βγει μια ποσοτική βαθμολογία γίνεται σύγκριση του χάρτη του ειδικού για την συγκεκριμένη δραστηριότητας με τον χάρτη του εκπαιδευόμενου. Τα λάθη, οι παραλήψεις και τα περιττά στοιχεία στον χάρτη του μαθητή κατηγοριοποιούνται σε μια από τις παραπάνω κατηγορίες. Κάθε κατηγορία έχει ένα ειδικό βάρος με το οποίο αθροίζεται σε ένα συνολικό βάρος και από αυτό βγαίνει μια βαθμολογία με την οποία αξιολογείται ποσοτικά η πορεία του εκπαιδευόμενου.

Εξ ορισμού, όταν ένας συγγραφέας κατασκευάζει μια δραστηριότητα υπολογίζονται αυτές οι τιμές για τις κατηγορίες των λαθών. Ανάλογα όμως με το είδος της δραστηριότητας που κατασκευάζει μπορεί να επιλέξει και να αλλάξει τα βάρη για κάποια από τα λάθη. Για παράδειγμα σε μια δραστηριότητα που έχει ως κύριο στόχο την εξ αρχής κατασκευή ενός εννοιολογικού χάρτη από τους εκπαιδευόμενους μπορεί να δοθεί μεγαλύτερο βάρος στην παράληψη μιας έννοιας από το να προσθέσει μια έννοια που δεν χρειάζεται.

Σε αντίθεση με τον συγγραφέα που μπορεί να μεταβάλλει τις τιμές των βαρών όλων των κατηγοριών λαθών για μια συγκεκριμένη δραστηριότητα, ο διαχειριστής μπορεί να μεταβάλλει τις τιμές για όλες τις δραστηριότητες.

2.2.4 Συνεργασία Συγγραφέων Δραστηριοτήτων Εννοιολογικής Χαρτογράφησης

Η τελευταία αλλά πολλή σημαντική λειτουργία του εργαλείου συγγραφής δραστηριοτήτων εννοιολογικής χαρτογράφησης είναι ότι επιτρέπει την συνεργασία μεταξύ των συγγραφέων δραστηριοτήτων έτσι ώστε με την ανταλλαγή απόψεων και γνώσεων να βελτιωθούν οι δραστηριότητες με σκοπό να πετύχουν καλύτερα τα προσδοκώμενα αποτελέσματα.

Όπως αναφέρθηκε ανωτέρω οι δραστηριότητες μπορούν να βρίσκονται σε μια από τις τρεις καταστάσεις: «Επεξεργασία», «Προβολή/Αξιολόγηση», «Οριστικοποίηση». Μια δραστηριότητα βρίσκεται στην κατάσταση «Επεξεργασία» κατά την δημιουργία της. Όταν ο συγγραφέας έχει τελειώσει με την δημιουργία της δραστηριότητάς του, μπορεί να αλλάξει την κατάσταση της, και να την μεταφέρει μέσα από το ειδικό μενού σε κατάσταση «Προβολή/Αξιολόγηση». Το στάδιο αυτό είναι ένα βήμα πριν την τελική διάθεση της δραστηριότητας προς εκπόνηση μέσα από το περιβάλλον του Compass. Για την διάθεση της δραστηριότητας στο Compass, θα πρέπει να μεταφερθεί από την κατάσταση «Προβολή/Αξιολόγηση» στην κατάσταση «Οριστικοποίηση».

Η κατάσταση «Προβολή/Αξιολόγηση», επιτρέπει τον σχολιασμό από τους άλλους συγγραφείς. Μέσα από την ειδική φόρμα, όλοι οι συγγραφείς μπορούν να στέλνουν τα σχόλια τους και να απαντούν σε αυτά. Η φόρμα έχει την μορφή συζήτησης (forum) και έτσι είναι εύκολη η διαλογική μορφή επικοινωνίας των συγγραφέων. Εκτός, όμως από τα παραπάνω υποστηρίζονται και σχόλια, κρυφά για όλους εκτός από τον δημιουργό της δραστηριότητας. Οι άλλοι συγγραφείς, έχουν την δυνατότητα προβολής, αλλά όχι επεξεργασίας μιας δραστηριότητας η οποία έχει διατεθεί προς αξιολόγηση.

3. Μελλοντικές Επεκτάσεις

Για την αξιολόγηση της λειτουργικότητας του περιβάλλοντος Compass έγινε πειραματική μελέτη σε μαθητές Γυμνασίου για πέντε εβδομάδες. Η συγκεκριμένη μελέτη απέδωσε χρήσιμα συμπεράσματα τα οποία χρησιμοποιήθηκαν για την βελτίωση του συστήματος. Επόμενος στόχος μας είναι να γίνει μια παρόμοια μελέτη αξιολόγηση για το εργαλείο συγγραφής δραστηριοτήτων Εννοιολογικής Χαρτογράφησης από έμπειρους και μη εκπαιδευτές διαφορετικών γνωστικών αντικειμένων έτσι ώστε να διερευνηθεί η χρηστικότητα, λειτουργικότητα και αποδοτικότητα του συστήματος σε πραγματικές δραστηριότητες μάθησης και εκπαίδευσης.

Αναφορές

1. Cañas, A., Hill, G., Carff, R., Suri, N., Lott, J., Gómez, G., Eskridge, T., Arroyo, M., & Carvajal, R. (2004). CmapTools: A knowledge modelling and sharing environment. In A. Cañas, J. Novak, & F. González (Eds.), *Concept Maps: Theory, Methodology, Technology, Proceedings of the First International Conference on Concept Mapping*, Pamplona, Spain.
2. Cimolino, L., Kay, J., & Miller, A. (2003). Incremental student modelling and reflection by verified concept-mapping. In *Supplementary Proceedings of AIED2003: Learner Modelling for Reflection*, 219-227.
3. Coffey, J. (2007). A meta-cognitive tool for courseware development, maintenance, and reuse. *Computers & Education*, 48(4), 548-566.
4. Coffey, J., & Cañas, A. (2003). LEO: A learning environment organizer to support computer-mediated instruction. *Journal for Educational Technology Systems*, 31(3), 275-290.
5. Conlon, T. (2006). Formative assessment of classroom concept maps: the Reasonable Fallible Analyser. *Journal of Interactive Learning Research*, 17(1), 15-36.
6. Gouli, E., Gogoulou, A., Papanikolaou, K., & Grigoriadou, M. (2004). COMPASS: An adaptive web-based concept map assessment tool. In A. Cañas, J. Novak, & F. González (Eds.), *Concept Maps: Theory, Methodology, Technology, Proceedings of the First International Conference on Concept Mapping*, Pamplona, Spain, available at <http://cmc.ihmc.us/CMC2004Programa.html>.
7. Hoefft, R., Jentsch, F., Harper, M., Evans, W., Bowers, C., & Salas, E. (2003). TPL-KATS-concept map: a computerized knowledge assessment tool. *Computers in Human Behavior*, 19, 653-657.
8. Hsieh, I.-L., & O'Neil, H. (2002). Types of feedback in a computer-based collaborative problem-solving group task. *Computers in Human Behavior*, 18, 699-715.
9. Novak, J., & Gowin, B. (1984). *Learning How to Learn*. New York: Cambridge University Press.
10. Rueda, U., Larrañaga, M., Arruarte, A. and Elorriaga, J. (2004). Applications of a concept mapping tool. In A. Cañas, J. Novak, & F. Gonzalez (Eds.), *Concept Maps: Theory, Methodology, Technology, Proceedings of the First International Conference on Concept Mapping*, Pamplona, Spain.

Σχεδιασμός και Μελέτη Επιδόσεων Καινοτόμου Σχήματος Διαμόρφωσης για Εφαρμογή σε Αρχιτεκτονικές Παθητικών Οπτικών Δικτύων Πρόσβασης

Ιωάννης Ανδρουλακακης¹

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών,
Τμήμα Πληροφορικής και Τηλεπικοινωνιών,
Πανεπιστημιούπολη, Ιλίσια, 15784, Αθήνα, Ελλάδα
e-mail : grad0690@di.uoa.gr

Περίληψη Οι εξελίξεις στο χώρο των οπτικών επικοινωνιών έχουν καταστήσει δυνατή τη χρήση της οπτικής ίνας και στο κομμάτι των δικτύων πρόσβασης κυρίως με την υλοποίηση των παθητικών οπτικών δικτύων. Στην αναφορά αυτή προτείνεται ένα καινοτόμο σχήμα διαμόρφωσης που βασίζεται στην ήδη γνωστή τεχνική της ορθογώνιας διαμορφωσης και με τη βοήθεια προγραμμάτων εξομοίωσης μελετώνται οι επιδόσεις του. Το σχήμα αυτό, το οποίο είναι κατάλληλο για εφαρμογή σε παθητικά οπτικά δίκτυα αυξάνει τη χωρητικότητα του δικτύου, δίνει τη δυνατότητα για ταυτόχρονη μεταφορά υπηρεσιών εικόνας και δεδομένων, καθώς και για υλοποίηση συνδρομητικών υπηρεσιών ψηφιακής εικόνας.

Λέξεις κλειδιά: ορθογώνια διαμόρφωση, παθητικά οπτικά δίκτυα, δίκτυα πρόσβασης, διαμόρφωση FSK, οπτικός διαμορφωτής

1 Εισαγωγή

Στα σύγχρονα τηλεπικοινωνιακά δίκτυα οι νέες υπηρεσίες που προστίθενται καθημερινά οδηγούν σε αύξηση των αναγκών σε εύρος ζώνης. Ιδιαίτερα η εισαγωγή των υπηρεσιών τύπου triple play services (TPS), μέσω των οποίων υπηρεσίες φωνής, ψηφιακής εικόνας βίντεο και δεδομένων υψηλής ταχύτητας μεταδίδονται από το ίδιο δίκτυο, δημιουργεί την ανάγκη ώστε το δίκτυο να διαθέτει κάποια ειδικά χαρακτηριστικά. Πιο συγκεκριμένα με τις υπηρεσίες TPS στο χρήστη πρέπει να παρέχονται ταχύτητες πολλών δεκάδων Mbps, οι οποίες θα αξιοποιηθούν για τις ήδη υπάρχουσες υπηρεσίες του διαδικτύου, ενώ το δίκτυο πρέπει να είναι σε θέση να παρέχει μηχανισμούς που θα εγγυώνται την ποιότητα της υπηρεσίας (Quality of service – QoS). Επιπλέον, το δίκτυο πρέπει να είναι συμβατό με τις παραδοσιακές υπηρεσίες τηλεφωνίας, που θα παρέχονται μαζί με τις υπηρεσίες φωνής που αξιοποιούν το πρωτόκολλο IP (Voice over IP). Τέλος, το δίκτυο πρέπει να παρέχει στο χρήστη ψηφιακή εικόνα βίντεο και συγκεκριμένα αρκετές δεκάδες από κανάλια

¹ Επιβλέποντες : Δημήτριος Συβρίδης και Παντελής Βελάνας

ποιοτητας υψηλής ευκρίνειας, μαζί με τις συνοδευτικές υπηρεσίες θέασης με συνδρομή (pay per view - PPV) και θέασης κατ' επιλογή (video on demand – VoD).

Από τα παραπάνω γίνεται σαφές ότι οι απαιτήσεις από ένα τηλεπικοινωνιακό δίκτυο γίνονται ολοένα και περισσότερες. Ωστόσο, το δίκτυο πρόσβασης εξακολουθεί να αποτελεί επιβραδυντικό παράγοντα για τις συνολικές επιδόσεις ενός δικτύου, αφού ακόμα και με την πρόοδο που έχει συντελεστεί στις τεχνικές μετάδοσης, οι ταχύτητες που επιτυγχάνονται σε αυτό είναι μικρές συγκρινόμενες με τις ταχύτητες που επιτυγχάνονται στα μητροπολιτικά δίκτυα και στα δίκτυα μεγάλων αποστάσεων [2]. Οι ήδη υλοποιημένες τεχνικές που χρησιμοποιούνται είτε έχουν σαν μέσο διάδοσης το συνεστραμμένο χάλκινο σύρμα (ADSL και VDSL), είτε την ατμόσφαιρα (802.11, 802.16, UMTS) λύνουν προσωρινά μόνο το πρόβλημα της χωρητικότητας και της ραγδαίας αύξησης του αριθμού των χρηστών, αλλά δεν μπορούν σε καμία περίπτωση να υποστηρίξουν πλήρως τις δυνατότητες ενός συστήματος υπηρεσιών TPS.

Προκειμένου να μειωθεί η υποβάθμιση που υφίσταται ένα δίκτυο συνολικά από τις χαμηλές ταχύτητες του δικτύου πρόσβασης, κρίνεται αναγκαία η χρήση της οπτικής ίνας και στο κομμάτι αυτό κάτι που επιτυγχάνεται με την τεχνική της ίνας στο σπίτι (fiber to the home – FTTH) και τις παραλλαγές της [3,4]. Μία πολλά υποσχόμενη λύση για πρακτική εφαρμογή σε δίκτυα που χρησιμοποιούν την τεχνική της ίνας στο σπίτι είναι τα παθητικά οπτικά δίκτυα (passive optical networks –PON) [2,3], τα οποία δεν χρησιμοποιούν ενεργά ηλεκτρονικά στοιχεία και με τον τρόπο αυτό μειώνουν σημαντικά το κόστος της υλοποίησης και λειτουργίας τους.

Μέχρι σήμερα έχουν προτυποποιηθεί τρεις διαφορετικές αρχιτεκτονικές παθητικών οπτικών δικτύων, οι οποίες έχουν ήδη αρχίσει να υλοποιούνται εμπορικά. Αυτές είναι το ευρείας ζώνης παθητικό οπτικό δίκτυο (broadband passive optical network – BPON) [3], το οποίο προτυποποιήθηκε από την διεθνή ένωση τηλεπικοινωνιών (International telecommunications union - ITU-T), το παθητικό οπτικό δίκτυο ταχύτητας της τάξης του gigabit (gigabit PON – GPON) [3], το οποίο προτυποποιήθηκε από τον ίδιο φορέα και το παθητικό οπτικό δίκτυο βασισμένο στο Ethernet (Ethernet-based passive optical network – EPON) [4], το οποίο προτυποποιήθηκε από το ινστιτούτο ηλεκτρολόγων και ηλεκτρονικών μηχανικών (international electrical and electronics engineers –IEEE).

Η χρήση βελτιωμένων τεχνολογιών όμως, δεν είναι ο μόνος τρόπος να αυξηθεί η χωρητικότητα ενός δικτύου πρόσβασης. Βελτίωση στη χωρητικότητα προκύπτει και από τη χρήση περισσότερο αποδοτικών τρόπων διαμόρφωσης της μεταδιδόμενης πληροφορίας. Για το σκοπό αυτό έχουν προταθεί μέθοδοι όπως τα σχήματα ορθογώνιας διαμόρφωσης φωτεινής έντασης / κλειδώματος μεταλλαγής συχνότητας IM/FSK [8,9] και ορθογώνιας διαμόρφωσης φωτεινής έντασης / διαφορικού κλειδώματος μεταλλαγής φάσης IM/DPSK [8,9]. Η ορθογώνια διαμόρφωση χρησιμοποιώντας το ίδιο φάσμα με ένα απλό σχήμα διαμόρφωσης μπορεί να προσφέρει ένα επιπλέον κανάλι για μετάδοση δεδομένων. Το γεγονός αυτό από μόνο του είναι ιδιαίτερα σημαντικό αφού μπορεί να συμβάλλει στην αύξηση της χωρητικότητας και συνεπώς στην κλιμάκωση (scalability) του αριθμού των χρηστών του δικτύου. Για το λόγο αυτό έχουν γίνει αρκετές ερευνητικές προσπάθειες, σχετικές με το θέμα [6,8,9].

Στη συγκεκριμένη εργασία ένα καινοτόμο σχήμα διαμόρφωσης που βασίζεται στην ορθογώνια διαμόρφωση IM/FSK υλοποιείται στην κατερχόμενη ροή

(downstream). Η διαμόρφωση FSK [5] επιτυγχάνεται με άμεση διαμόρφωση ενός λέιζερ καταναμημένης ανατροφοδότησης (DFB) [10] και η διαμόρφωση IM με τη χρήση ενός εξωτερικού διαμορφωτή τύπου Mach Zehnder [2,5]. Με το σήμα που είναι διαμορφωμένο κατά IM μεταφέρονται δεδομένα στους χρηστές με πολυπλεξία με διαίρεση χρόνου (TDMA) και συνολικό ρυθμό 10 Gbps και με το σήμα που είναι διαμορφωμένο κατά FSK μεταφέρεται εικόνα βίντεο προς διακριτές ομάδες χρηστών με συνολικό ρυθμό 1.25 Gbps με τον τρόπο που περιγράφεται παρακάτω.

Με το μηχανισμό που προτείνεται οι χρήστες χωρίζονται σε δύο διακριτές ομάδες. Όλοι έχουν πρόσβαση στην πληροφορία που είναι διαμορφωμένη κατά IM, αλλά ανάλογα με την ομάδα στην οποία ανήκουν λαμβάνουν διαφορετική πληροφορία που αφορά στην ψηφιακή εικόνα. Ενδεικτικά, η μία ομάδα λαμβάνει απλή ροή εικόνας βίντεο και η άλλη εικόνα βίντεο που περιέχει ένα επιπλέον αριθμό προγραμμάτων, για τα οποία έχει πληρώσει κάποιου είδους συνδρομή [7]. Εναλλακτικά, οι ομάδες των χρηστών μπορούν να διαχωριστούν και με άλλους τρόπους, ανάλογα με τις ανάγκες του συστήματος. Η αρχιτεκτονική που μελετάται πέρα από τη χρήση της σε συνδρομητικές υπηρεσίες, συμβάλλει στην αύξηση της χωρητικότητας του δικτύου, στην κλιμάκωση των χρηστών και στην δυνατότητα εισαγωγής νέων ευρυζωνικών υπηρεσιών σε ένα δίκτυο. Τα αποτελέσματα που προέκυψαν από τα προγράμματα εξομοίωσης παρουσιάζουν εξαιρετικό ερευνητικό ενδιαφέρον μια και επιβεβαιώνουν τη δυνατότητα χρήσης της αρχιτεκτονικής για τους σκοπούς που προαναφέρθηκαν. Μέρος από τα αποτελέσματα αυτά έχουν δημοσιευθεί με την αναφορά P.Velanas, Y. Androulakakis, A.Bogris, D.Syvridis, “ A novel transmitter based on orthogonal modulation schemes for future passive networks”, Optical Fiber Communication Conference, Vol. JThA102, San Diego, USA, Febuary 2008

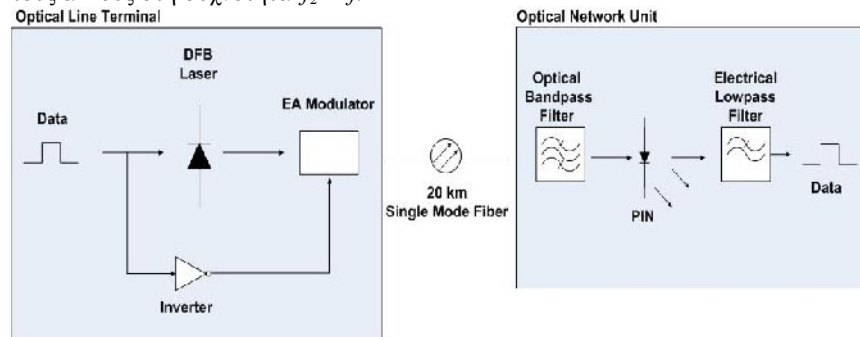
2 Περιγραφή της προτεινόμενης αρχιτεκτονικής

Όπως είναι γνωστό ένα σήμα FSK παράγεται άμεσα με την εφαρμογή ενός σταθερού ρεύματος πόλωσης, πάνω από το κατώφλι, σε ένα λέιζερ DFB και ταυτόχρονη εφαρμογή ενός μικρού ρεύματος διαμόρφωσης. Με τον τρόπο αυτό έχουμε την εμφάνιση του τόνου «0» σε μια συχνότητα f_1 και την εμφάνιση του τόνου «1» σε μια συχνότητα $f_1 + \Delta f$. Το μπλοκ διαγραμμα ενός τέτοιου διαμορφωτή αλλά και του αντιστοιχού δέκτη που χρησιμοποιείται για τη φώραση φαίνεται στην εικόνα 1.

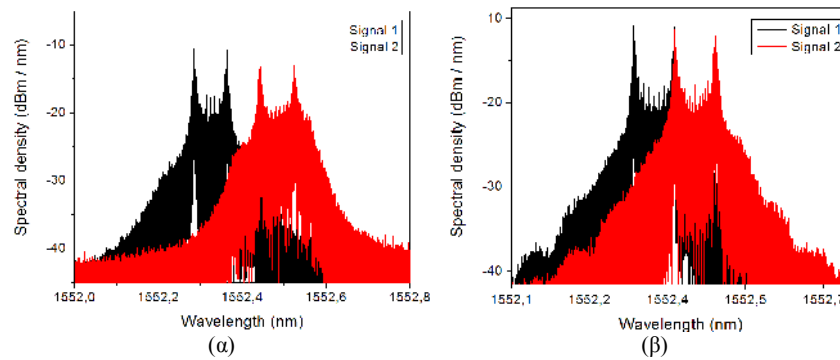
Με τη μέθοδο που προτείνεται, το ρεύμα διαμόρφωσης στο διαμορφωτή παραμένει σταθερό, αλλά μεταβάλλεται το ρεύμα πόλωσης, ώστε ο τόνος «1» να μεταδίδεται στις συχνότητες $f_1 + \Delta f$ και $f_2 + \Delta f$, όπου $f_2 > f_1$. Είναι σαφές ότι ο τόνος μηδέν δεν βρίσκεται σταθερά σε μια συχνότητα και μετατοπίζεται αντίστοιχα από τη συχνότητα f_1 στη συχνότητα f_2 . Η διαδικασία αλλαγής του ρεύματος πόλωσης του λέιζερ είναι τέτοια ώστε στη μίση χρονοθυρίδα το σήμα του FSK να εμφανίζει τον τόνο «1» στη συχνότητα $f_1 + \Delta f$ και να μεταδίδει δεδομένα στη μια ομάδα χρηστών και στην άλλη μίση να εμφανίζει τον τόνο «1» στη συχνότητα $f_2 + \Delta f$ και να μεταδίδει στην άλλη ομάδα χρηστών. Το φάσμα του σήματος σε αυτή την περίπτωση έχει τη μορφή της εικόνας 2α.

Κατά την άμεση διαμόρφωση με FSK εμφανίζεται μια αυξομείωση στο πλάτος του διαμορφωμένου σήματος, η οποία είναι δυνατόν να υποβαθμίσει την ποιότητα

της επικοινωνίας. Προκειμένου το φαινόμενο αυτό να εξαλειφθεί χρησιμοποιείται μετά την έξοδο του DFB ένας διαμορφωτής ηλεκτροαπορρόφησης πολωμένος με την κατάλληλη τάση. Το σήμα μετά τη διάδοση του από μια ίνα 20 χιλιομέτρων, μήκος που είναι ενδεικτικό για τα δίκτυα πρόσβασης, φτάνει στον δέκτη όπου μετατρέπεται σε ηλεκτρικό μετά από οπτικό φιλτραρισμα του τόνου “1” και διέλευση από μια φωτοδίοδο PIN. Είναι προφανές, ότι οι δέκτες στα ONU είναι δύο τύπων και έχουν διαφορετική συχνότητα αποκόπής στο φίλτρο τους, με αυτούς που αντιστοιχούν σε χρήστες της μίας ομάδας να λαμβάνουν την πληροφορία στη συχνότητα $f_1 + \Delta f$ και τους άλλους στη συχνότητα $f_2 + \Delta f$.



Εικ. 1. Το block διάγραμμα του πομπού και του δέκτη που χρησιμοποιείται στην αρχιτεκτονική του απλού FSK.

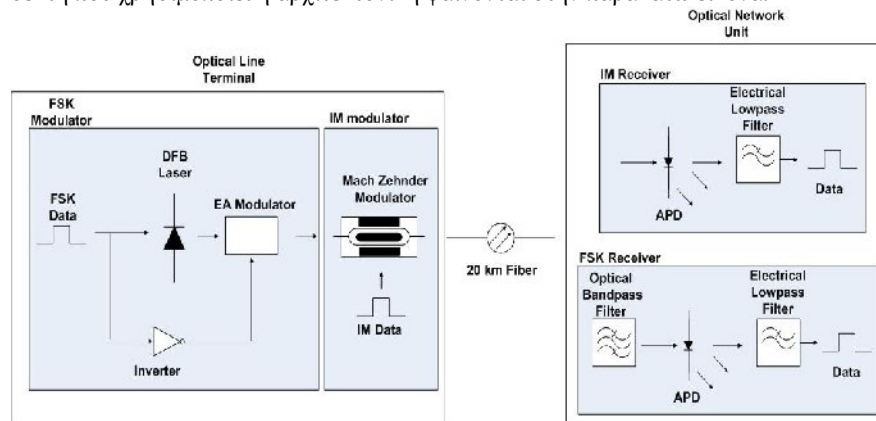


Εικ. 2. Στα αριστερά εικονίζεται το φάσμα που παράγεται από το διαμορφωτή απλού FSK και δεξιά εικονίζεται το φάσμα που παράγεται από το διαμορφωτή IM/FSK.

Στην περίπτωση της ορθογώνιας διαμόρφωσης IM/FSK ο μηχανισμός που εξασφαλίζει το διαχωρισμό σε ομάδες χρηστών είναι παρόμοιος με αυτόν που ήδη περιγράφηκε. Όπως και στην περίπτωση του απλού FSK, με κατάλληλη εφαρμογή σταθερού ρεύματος πόλωσης πάνω από το κατώφλι σε ένα λέιζερ DFB και ταυτόχρονη εφαρμογή ενός μικρού ρεύματος διαμόρφωσης, έχουμε την εμφάνιση του τόνου «0» του σήματος σε μια συχνότητα f_1 και την εμφάνιση του τόνου «1» σε μια συχνότητα $f_1 + \Delta f$. Με τη μέθοδο που προτείνεται, το ρεύμα πόλωσης στο διαμορφωτή

παραμένει σταθερό, αλλά μεταβάλλεται το ρεύμα διαμόρφωσης, ώστε ο τόνος «0» να εξακολουθήσει να μεταδίδεται στο f_i , αλλά ο τόνος «1» να μεταδίδεται πλέον στο $f_i - \Delta f$. Η διαδικασία αλλαγής του ρεύματος διαμόρφωσης του λέιζερ και η κατάλληλη αντιστροφή της ακολουθίας της πληροφορίας που απαιτείται είναι τέτοια ώστε στη μίση χρονοθυρίδα ο τόνος «1» του σήματος FSK να βρίσκεται στη συχνότητα $f_i + \Delta f$ και να μεταδίδει δεδομένα στη μια ομάδα χρηστών και στην άλλη μίση να βρίσκεται στη συχνότητα $f_i - \Delta f$ και να μεταδίδει στην άλλη ομάδα χρηστών. Ο τόνος «0» του σήματος FSK βρίσκεται συνεχώς στη συχνότητα f_i .

Η χρήση του φάσματος γίνεται περισσότερο αποδοτική αφού το ήδη διαμορφωμένο σήμα με FSK επαναδιαμορφώνεται κατά οπτική ένταση με διαφορετική πληροφορία με τη βοήθεια ενός εξωτερικού διαμορφωτή Mach Zehnder. Το φάσμα μετά από αυτή τη διαδικασία φαίνεται και στην εικόνα 2β. Το διαμορφωμένο σήμα διαδίδεται και η φάραση του στο δέκτη γίνεται αφού διαχωριστεί σε δύο όμοια σήματα από ένα παθητικό διαχωριστή. Το ένα οδηγείται σε ένα δέκτη FSK, ο οποίος λειτουργεί με την ίδια φιλοσοφία όπως και πριν και το άλλο σε έναν που διαθέτει μια φωτοδίοδο APD, η οποία κάνει άμεση αποδιαμόρφωση στο σήμα που είναι διαμορφωμένο κατά IM. Το διάγραμμα block του πομπού και του δέκτη που χρησιμοποιεί η αρχιτεκτονική φαίνονται στην παρακάτω εικόνα.



Εικ. 3. Το block διάγραμμα του πομπού και του δέκτη που χρησιμοποιεί η αρχιτεκτονική του ορθογώνιου IM/FSK

3 Αποτελέσματα

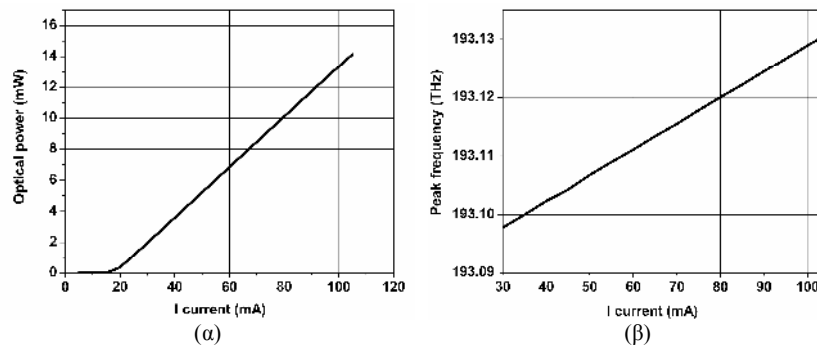
Η διαμόρφωση τόσο κατά FSK, όσο και κατά IM/FSK πραγματοποιείται με χρήση ενός λέιζερ DFB, του οποίου οι κύριες παράμετροι φαίνονται στον πίνακα 1 και οι χαρακτηριστικές του, όπως αυτές προκύπτουν από την εξομοίωση στην εικόνα 4. Για τη λήψη των αποτελεσμάτων και την αξιολόγηση της αρχιτεκτονικής χρησιμοποιήθηκε η πλατφόρμα εξομοίωσης οπτικών δικτύων Vpi, η οποία δίνει τη δυνατότητα να εξάγονται ιδιαίτερα ακριβή αποτελέσματα.

TABLE I
DFB LASER CHARACTERISTICS

| | |
|-------------------------------|--|
| Laser Emission Wavelength | 1550 nm |
| Laser Chip Length | 300 μm |
| Active Region Width | 3 μm |
| Active Region Thickness | 0.2 μm |
| Linear Material Gain Coeff. | $3.3 \times 10^{20} \text{ m}^{-2}$ |
| Transp. Carrier Density | $1.5 \times 10^{24} \text{ m}^{-3}$ |
| Confinement Factor | 0.3 |
| Group Effective Index | 4 |
| Material Linew. Enhanc. Fact. | 9 |
| Differential Refractive Ind. | $-1.63 \times 10^{-26} \text{ m}^{-3}$ |
| Left/Right Facet Reflect. | 0.3 |
| Photon lifetime | $3 \times 10^{-12} \text{ s}$ |
| Carrier lifetime | $2.1 \times 10^{-9} \text{ s}$ |
| Bimol. Recomb. Coeff. | $10^{-16} \text{ m}^3/\text{s}$ |
| Auger Recomb. Coeff. | $10^{-41} \text{ m}^6/\text{s}$ |
| Spont. Emis. Factor | 10^{-4} |
| Pop. Inversion Parameter | 2 |
| Nonlinear Gain Coefficient | $3 \times 10^{-23} \text{ m}^{-3}$ |
| Gain Suppresion Coeff. | 0.03 1/W |

Πίνακας 1. Παράμετροι του χρησιμοποιούμενου λέιζερ DFB.

Στήν εικόνα 4α φαίνεται καθαρά η γραμμική αύξηση της οπτικής ισχύος ανάλογα με το ρεύμα πόλωσης. Το κατώφλι του λέιζερ βρίσκεται στα 22 mA περίπου, ενώ για αύξηση της οπτικής ισχύος από 7.5 σε 15 mW απαιτείται αύξηση του ρεύματος έγχυσης κατά 50 mA. Αντίστοιχα στο σχήμα 4β φαίνεται η γραμμική εξάρτηση της οπτικής συχνότητας εκπομπής από το ρεύμα έγχυσης. Συγκεκριμένα προκειμένου να επιτευχθεί αύξηση της οπτικής συχνότητας από τα 193.11 THz στα 193.12 THz απαιτείται αύξηση του ρεύματος κατά 22 mA.

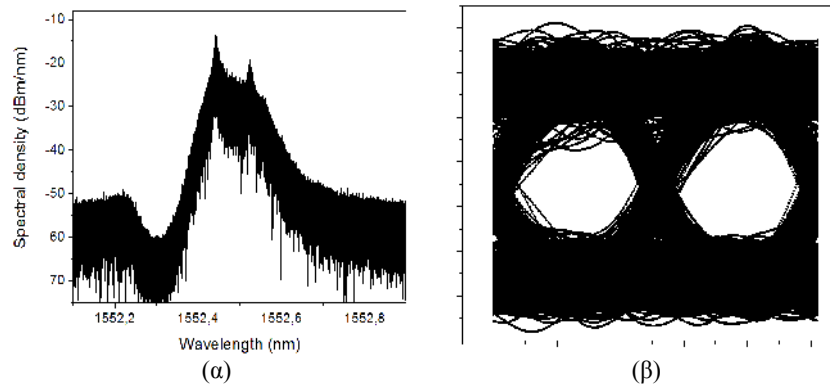


Εικ. 4. Χαρακτηριστικές του λέιζερ

Το φάσμα που διαδίδεται τελικά στην ίνα έχει τη μορφή της εικ. 2α για την περίπτωση του απλού FSK. Στο λέιζερ εφαρμόζεται ρεύμα πόλωσης 35 mA και ρεύμα διαμόρφωσης 22 mA ώστε ο τόνος “1” του σήματος στη μισή χρονοθυρίδα

όπου μεταδίδεται πληροφορία για τη μία ομάδα χρηστών να βρίσκεται στη συχνότητα 193.11 THz και ο τόνος «0» στη συχνότητα 193.1 THz. Στην άλλη μισή όπου μεταδίδεται πληροφορία που προορίζεται για τη δεύτερη ομάδα εφαρμόζεται ρεύμα πόλωσης 80 mA και ρεύμα διαμόρφωσης 22 mA ώστε ο τόνος «1» να βρίσκεται στη συχνότητα 193.13 THz και ο τόνος «0» στη συχνότητα 193.12 THz.

Ο ρυθμός μεταδοσης είναι 1.25 GHz και από τα δεδομένα της εξομοίωσης ο μέγιστος αριθμός των ONU που μπορούν να εξυπηρετηθούν είναι 256. Το σήμα διαδίδεται μέσα από μια απλή μονότροπη ίνα μήκους 20 χιλιομέτρων, ενώ στο δέκτη της κάθε ομάδας για την ανάκτηση του τόνου «1», στον οποίο είναι αποθηκευμένη η πληροφορία, χρησιμοποιείται συστοιχία φίλτρων, ενός Fabry-Perot και ενός Fiber Bragg Grating (FBG) η οποία έχει σαν αποτέλεσμα την υποβάθμιση του τόνου «0» κατά 12 dB περίπου. Στην εικόνα 5α φαίνεται το φάσμα στο δέκτη μετά τη διέλευση του σήματος από τη συστοιχία των φίλτρων για τη μια ομάδα χρηστών και στην 5β το διάγραμμα οφθαλμού που προκύπτει μετά τη φώραση του ηλεκτρικού σήματος. Για τις ήδη προαναφερθείσες παραμέτρους η ισχύς που φτάνει στο δέκτη είναι 2.47 μ W και ο ρυθμός σφάλματος bit (BER) είναι $1.88 \cdot 10^{-11}$. Αντίστοιχα αποτελέσματα λαμβάνονται και για την δεύτερη ομάδα χρηστών.

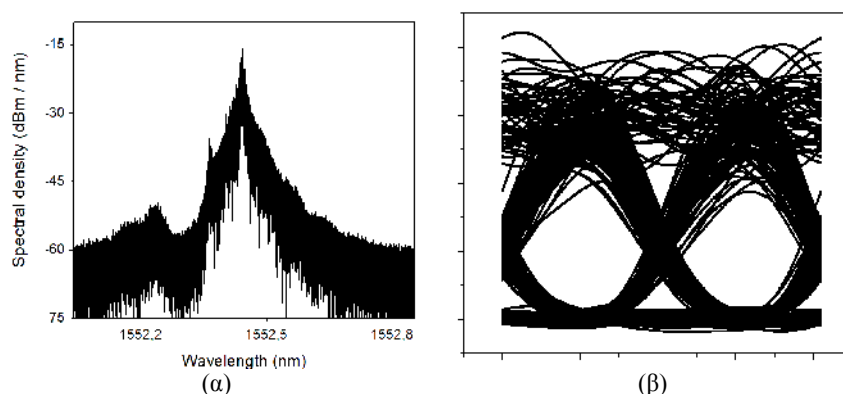


Εικ. 5. Φάσμα στο δέκτη και διάγραμμα οφθαλμού για τη μία ομάδα χρηστών.

Το φάσμα που διαδίδεται στην ίνα έχει τη μορφή της εικ. 2β για την περίπτωση του ορθογώνιου IM/FSK. Αρχικά στο λέιζερ του διαμορφωτή εφαρμόζεται ρεύμα πόλωσης 80 mA, ώστε να παραχθεί ο τόνος «0» του σήματος FSK στη συχνότητα 193.12 THz. Στη μισή χρονική διάρκεια της χρονοθυρίδας, όταν το bit που μεταδίδεται είναι «1», ασκείται αντίστροφη τάση διαμόρφωσης, ώστε το συνολικό ρεύμα που εφαρμόζεται στο λέιζερ τότε να έχει την τιμή 58 mA και ο τόνος «1» του σήματος FSK να βρίσκεται στη συχνότητα 193.11 THz. Στην άλλη μισή χρονοθυρίδα, στην οποία μεταδίδονται δεδομένα για τους χρηστές της δεύτερης ομάδας, εφαρμόζεται στο λέιζερ τάση διαμόρφωσης τέτοια ώστε όταν το μεταδιδόμενο bit είναι «1» το συνολικό ρεύμα που εγγεύεται στο λέιζερ να είναι 102 mA και ο τόνος «1» να βρίσκεται στην οπτική συχνότητα 193.13 THz.

Η πληροφορία που είναι διαμορφωμένη κατά FSK έχει ρυθμό 1.25 Gbps, όπως και πριν, και η πληροφορία που είναι διαμορφωμένη κατά IM έχει ρυθμό 10 Gbps.

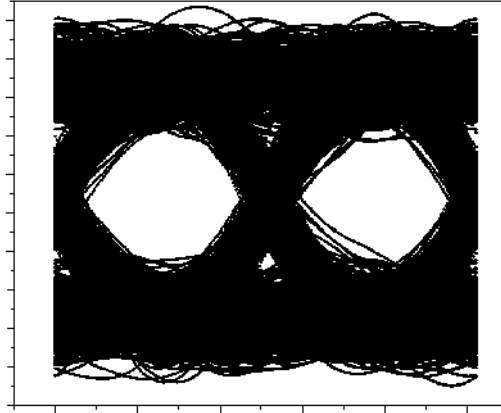
Στην περίπτωση αυτή, ωστόσο, ο αριθμός των ONU που μπορούν να χρησιμοποιηθούν είναι 64 λόγω της αλληλεπίδρασης των σημάτων FSK και IM. Το σήμα διαδίδεται και πάλι σε μήκος 20 χιλιομέτρων, αλλά λόγω του υψηλού ρυθμού του σήματος IM εμφανίζονται φαινόμενα διασποράς τα οποία αντιμετωπίζονται με τη χρήση ίνας μετατοπισμένης διασποράς (DCF). Στο δέκτη του FSK χρησιμοποιείται και εδώ συστοιχία φίλτρων κατάλληλα ρυθμισμένων ώστε να είναι δυνατή η φώραση των σημάτων από κάθε ομάδα χρηστών. Στην εικ. 6 παρατίθενται το φάσμα στο δέκτη του σήματος FSK καθώς και το διάγραμμα οφθαμού για την ομάδα που λαμβάνει δεδομένα στη συχνότητα 193.11 THz. Αντίστοιχα συμπεράσματα μπορούν να εξαχθούν και για την άλλη ομάδα. Η ισχύς που φτάνει στο δέκτη είναι $3.81 \mu\text{W}$ και ο ρυθμός σφάλματος $\text{bit } 1.5 \cdot 10^{-10}$



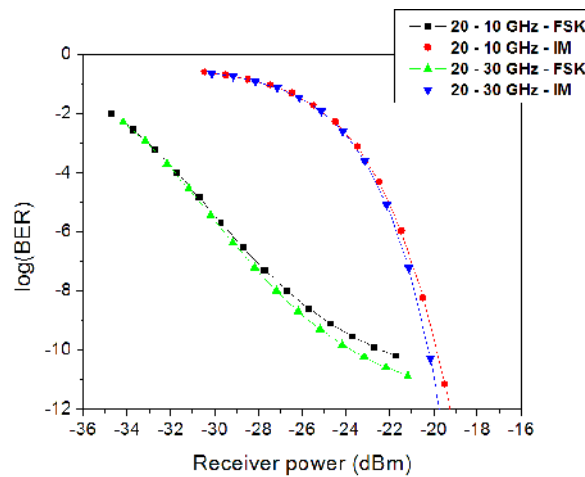
Εικ. 6. Φάσμα στο δέκτη και διαγραμμα οφθαμού για τη μία ομάδα χρηστών (σημα FSK).

Στό δέκτη του IM σε αντίθεση με την περίπτωση του FSK, όπου ένας δεδομένος δέκτης λαμβάνει οπτικό σήμα κατά το μίσο της χρονοθυρίδας, το σήμα που μεταδίδεται λαμβάνεται καθ' ολη τη διάρκεια της χρονοθυρίδας. Συνεπώς στη διαμόρφωση αυτή η έννοια της ομάδας δεν υπάρχει. Στην εικόνα 7 εικονίζεται το διάγραμμα οφθαμού για τη συγκεκριμένη διαμόρφωση. Η λαμβανόμενη ισχύς στο δεκτή είναι $9 \mu\text{W}$ και ο ρυθμός σφάλματος $\text{bit } 5 \cdot 10^{-11}$. Ανακεφαλαιώνοντας, το σύστημα ορθογώνιας διαμόρφωσης μπορεί να χρησιμοποιηθεί για ψηφιακή μετάδοση δεδομένων σε 64 ONU με ρυθμούς 10 Gbps για τη διαμόρφωση IM και 1.25 Gbps για τη διαμόρφωση FSK σε αποστάσεις 20 χιλιομέτρων.

Στην εικόνα 8 αναπαριστάται ο ρυθμός σφάλματος bit σε συνάρτηση με την ισχύ που λαμβάνεται στους δυο δεκτές. Είναι προφανές ότι για τη φώραση του FSK απαιτείται λιγότερη ισχύς στο δεκτή σε σχέση με το IM. Επιπλέον η υποβάθμιση όταν η άμεση διαμόρφωση του λέιζερ στον οπτικό διαμορφωτή γίνεται μεταξύ των συχνοτήτων 193.12 και 193.11 THz είναι λίγο μεγαλύτερη από την περίπτωση που η διαμόρφωση γίνεται μεταξύ των συχνοτήτων 193.12 και 193.13 THz. Ωστόσο η παραπάνω διαφορά δεν επηρεάζει τις δυνατότητες του συστήματος, όπως αυτές περιγράφηκαν παραπάνω.



Εικ. 7. Διάγραμμα οφθαλμού για το σήμα που είναι διαμορφωμένο κατά IM



Εικ. 8. Πιθανότητα σφάλματος bit σε συνάρτηση με τη λαμβανόμενη ισχύ στο δέκτη

4 Συμπεράσματα

Στη συγκεκριμένη εργασία έγινε ανάλυση ενός καινοτόμου σχήματος διαμόρφωσης που βασίζεται στην ορθογώνια διαμόρφωση. Το συγκεκριμένο σχήμα είναι εξαιρετικά απλό στην υλοποίηση και μπορεί να χρησιμοποιηθεί με λίγες τροποποιήσεις στα ήδη υπάρχοντα παθητικά οπτικά δίκτυα, αυξάνοντας τη χωρητικότητά τους. Επίσης μπορεί να χρησιμοποιηθεί για την υλοποίηση συνδρομητικών υπηρεσιών, και ιδιαίτερα για επιλεκτική εκπομπή ψηφιακής εικόνας

βίντεο. Τα αποτελέσματα της εξομοίωσης επιβεβαιώνουν ότι το προτεινόμενο σχήμα είναι κατάλληλο για χρήση στα συνεχώς αναπτυσσόμενα παθητικά οπτικά δίκτυα, τα οποία είναι τα πλέον κατάλληλα για να υποστηρίξουν τα πολλά υποσχόμενα δίκτυα υπηρεσιών TPS.

Αναφορές

1. P.Velanas, Y. Androulakis, A.Bogris, D.Syvridis,“ A novel transmitter based on orthogonal modulation schemes for future passive networks”, Optical Fiber Communication Conference, Vol. JThA102, San Diego, USA, Febuary 2008
2. R. Ramaswami, K.N. Sivarajan, *Optical Networks – A practical perspective*, Morgan Kaufmann, 2002
3. P.E. Green, *Fiber to the home. The new empowerment*, Wiley-Interscience, 2006
4. G. Kramer, *Ethernet passive optical networks*, McGraw-Hill, 2005
5. G.P. Agrawal, *Lightwave technology telecommunication systems*, Wiley-Interscience, 2005
6. N. Skarmoutsos, P. Velanas, D. Syvridis, and T. Sphicopoulos, “Capacity enhancement of passive optical networks using orthogonal modulation schemes”, *Journal of optical networking*, vol. 6, No 7, Jul. 2007, pp. 925-936
7. M. Khanal, C.J. Chae, and R.S. Tucker, “Selective Broadcasting of Digital Video Signals Over a WDM Passive Optical Network”, *IEEE Photonics Technology Letters*, vol. 17, No. 9, Sep. 2005, pp. 1992-1995
8. N. Chi, B. Carlsson, P. V. Holm-Nielsen, J. Zhang, C. Peucheret, and I. T. Monroy, “Numerical investigation of single channel transmission of the orthogonally modulated signals”, *STOLAS Project*, Apr. 2003
9. N. Chi, J. Zhang, P. V. Holm-Nielsen, B. Kozicki, C. Peucheret, I. T. Monroy, S. O'Duill, “Experimental results on single-channel transmission of orthogonally modulated signals”, *STOLAS Project*, Jun. 2003
10. R.S. Vodhanel, A. F. Elrefaie, M.Z. Iqbal, R.E. Wagner, J.L. Gimlett, and S. Tsuji, “Performance of directly modulated DFB Lasers in 10 Gb/s ASK, FSK, and DPSK Lightwave systems, *Journal of Lightwave technology*, vol.8, No.9, Sep. 1990, pp. 1379-1386

Επίλυση Αλγεβρικών Συστημάτων Μικρής Διάστασης στους Πραγματικούς

Δημήτρης Διώχνος

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών,
Μετ/κό πρόγραμμα Λογικής, Θεωρίας Αλγορίθμων και Υπολογισμού

Περίληψη

Ασχολούμαστε με επίλυση καλώς ορισμένων πολυωνυμικών συστημάτων με ακέραιους συντελεστές. Χρησιμοποιούμε πολυωνυμικές ακολουθίες υπολοίπων και τον αλγόριθμο του Sturm. Οι λύσεις των μεθόδων μας είναι πραγματικοί αλγεβρικοί αριθμοί που αναπαρίστανται υπό μορφή διαστημάτων απομόνωσης. Το προηγούμενο φράγμα με αυτή την αναπαράσταση ήταν $\tilde{O}_B(N^{30})$ και το βελτιώνουμε σε $\tilde{O}_B(N^{12})$. Τέλος, υλοποιήσαμε και τις τρεις προτεινόμενες μεθόδους σε MAPLE και παραθέτουμε πειράματα για την αποδοτικότητά τους σε σχέση με άλλες δημοφιλείς υλοποιήσεις.

Λέξεις κλειδιά: επίλυση στους πραγματικούς, πολυωνυμικά συστήματα, αλγόριθμος Sturm, διαστήματα απομόνωσης.

1 Εισαγωγή

Το πρόβλημα επίλυσης καλώς ορισμένων αλγεβρικών συστημάτων είναι καίριας σημασίας. Οι περισσότεροι αλγόριθμοι ασχολούνται με τη γενική περίπτωση ή αναζητούν λύσεις σε κλειστά αλγεβρικά σώματα [3, 16, 21, 22, 20]. Η παρούσα διπλωματική βασίζεται στα αποτελέσματα [5, 6] και επικεντρώνεται στην επίλυση στους πραγματικούς σε διμετάβλητα συστήματα. Έτσι βρίσκουμε ακριβή φράγματα πολυπλοκότητας γι' αυτή την περίπτωση και μελετούμε διάφορους αλγορίθμους στην πράξη. Οι αλγόριθμοι είναι επέκταση των αντίστοιχων μεθόδων στο [31]. Σχετικές εργασίες είναι [15, 9, 23]. Εναλλακτικές προσεγγίσεις βρίσκονται στα [19, 26]. Τέλος, ο προσδιορισμός της τοπολογίας μιας πραγματικής αλγεβρικής καμπύλης πρέπει να υπολογίσει εν κατακλείδι τις πραγματικές λύσεις ενός συστήματος δύο εξισώσεων σε δύο μεταβλητές, δείτε [2, 10, 34, 13, 1].

2 Προκαταρκτικά

Σε ότι ακολουθεί, με O_B εννοούμε πολυπλοκότητα bit και με \tilde{O}_B αγνοούμε (πολυ)-λογαριθμικούς παράγοντες. Για $f \in \mathbb{Z}[y_1, \dots, y_k, x]$, το $\deg(f)$ δείχνει το συνολικό βαθμό, ενώ το $\deg_x(f)$ δηλώνει το βαθμό του πολυωνύμου ως προς x . Με $\mathcal{L}(f)$ φράσσουμε το δυαδικό μήκος των συντελεστών του f (συμπεριλαμβανοντας ένα bit για το πρόσημο). Υποθέτουμε $\mathcal{L}(\deg(f)) = \mathcal{O}(\mathcal{L}(f))$.

*Επιβλέπων: Ιωάννης Ζ. Εμίρης. Συνεργασία με τον Ηλία Τσιγαρίδα.

2.1 Πραγματικοί Αλγεβρικοί Αριθμοί

Επιλέγουμε να αναπαραστήσουμε τους πραγματικούς αλγεβρικούς αριθμούς $\alpha \in \mathbb{R}_{alg}$ με αναπαράσταση διαστημάτων απομόνωσης (*isolating interval representation*). Η αναπαράσταση περιλαμβάνει ένα χωρίς-τετράγωνα (square-free) πολυώνυμο f το οποίο έχει ρίζα το α και ένα διάστημα με άκρα ρητούς το οποίο περιέχει το α και καμία άλλη ρίζα. Αν α η μοναδική ρίζα του f στο διάστημα $\mathcal{J} = [I_L, I_R]$, όπου $I_L, I_R \in \mathbb{Q}$, το συμβολίζουμε ως: $\alpha \simeq [f, \mathcal{J}] = [f, [I_L, I_R]]$.

2.2 Πολυωνυμικές Ακολουθίες Υπολοίπων

Καίρια σημασία στις μεθόδους μας έχει ο υπολογισμός του MKΔ δύο πολυωνύμων. Στις εφαρμογές ενδιαφερόμαστε για τις ρίζες του MKΔ. Αρκεί λοιπόν να υπολογιστεί ο MKΔ μέχρι ομοιότητα. Έτσι, χρησιμοποιούμε ακολουθίες υπολοίπων που μοιάζουν με την ακολουθία που προκύπτει από την ψευδο-ευκλείδεια διαίρεση των δύο πολυωνύμων. Οι παραλλαγές που χρησιμοποιούνται στην πράξη είναι οι *προσημασμένες ακολουθίες υπολοίπων* (βλ. [2, 33, 35] και αναφορές εκεί). Εδώ ασχολούμαστε με προσημασμένες Subresultant και Sturm-Habicht ακολουθίες ($\mathbf{SR}(f, g)$ και $\mathbf{StHa}(f, g)$ αντίστοιχα). Έτσι υπολογίζουμε μια ακολουθία που μοιάζει με την $R_0 = f, R_1 = g, R_2 = -\text{prem}(f, g), \dots, R_k = -\text{prem}(R_{k-2}, R_{k-1})$, όπου $\text{prem}(R_i, R_{i+1})$ το υπόλοιπο της αντίστοιχης ευκλείδειας ψευδο-διαίρεσης και $\text{prem}(R_{k-1}, R_k) = 0$. Στην περίπτωση μας, το πολυώνυμο g είναι η παράγωγος του f . βλ. [11, 2]. Ακολουθούν τα πιο σημαντικά αποτελέσματα σχετικά με τον υπολογισμό και την αποτίμηση αυτών των ακολουθιών. Με $\mathbf{sr}(f, g)$ συμβολίζουμε την ακολουθία των πρωτεύοντων συντελεστών υποαπαλοιφουσών (*principal subresultant coefficients*), με $\mathbf{SRQ}(f, g)$ τη μπότα πηλίκου (quotient boot) και με $\mathbf{SR}(f, g; a)$ την ακολουθία αποτιμημένη πάνω στο $a \in \mathbb{Q}$.

Πρόταση 2.1. [17, 18, 27] Έστω $p \geq q$. Η $\mathbf{SR}(f, g)$ υπολογίζεται σε χρόνο $\tilde{O}_B(p^2 q \tau)$ με $\mathcal{L}(\mathbf{SR}_j(f, g)) = \mathcal{O}(p \tau)$. Η μπότα πηλίκου, οποιοδήποτε πολυώνυμο στην $\mathbf{SR}(f, g)$, η απαλοιφούσα και ο MKΔ υπολογίζονται σε χρόνο $\tilde{O}_B(p q \tau)$.

Λήμμα 2.2. [17, 27] Έστω $p \geq q$. Μπορούμε να υπολογίσουμε την $\mathbf{SR}(f, g; a)$, όπου $a \in \mathbb{Q} \cup \{\pm\infty\}$ και $\mathcal{L}(a) = \sigma$, σε χρόνο $\tilde{O}_B(p q \tau + q^2 \sigma + p^2 \sigma)$, όπου $\tau = \max\{\mathcal{L}(f), \mathcal{L}(g)\}$. Αν το $f(a)$ γνωστό, τότε το φράγμα γίνεται $\tilde{O}_B(p q \tau + q^2 \sigma)$.

Ορισμός 2.3. Έστω L μια λίστα πραγματικών αριθμών. Με $\text{VAR}(L)$ συμβολίζουμε το πλήθος (πιθανώς τροποποιημένων, δείτε [2, 11]) εναλλαγών προσήμου.

Πόρισμα 2.4. Για οποιαδήποτε f, g , το $\text{VAR}(\mathbf{SR}(f, g; a))$ υπολογίζεται σε χρόνο $\tilde{O}_B(p q \tau + \min\{p, q\}^2 \sigma)$, δεδομένου πως το πρόσημο $\text{sign}(f(a))$ είναι γνωστό.

2.3 Πολυώνυμο μίας μεταβλητής

Οι μέθοδοι της ενότητας 3 στηρίζονται στην επίλυση πολυωνύμων μιας μεταβλητής στους πραγματικούς. Εδώ θα αναφέρουμε τα σημαντικότερα αποτελέσματα τα οποία χρειαζόμαστε για τη συνέχεια. Για περισσότερα δείτε [8].

Πρόταση 2.5 (Αλγόριθμος Sturm). [7, 8] Έστω $f \in \mathbb{Z}[x]$ με βαθμό p και $\mathcal{L}(f) \leq \tau_f$. Μπορούμε να υπολογίσουμε τις πραγματικές ρίζες και τις πολλαπλότητες του f υπό μορφή διαστημάτων απομόνωσης σε χρόνο $\tilde{O}_B(p^6 + p^4 \tau_f^2)$. Τα άκρα των διαστημάτων έχουν δυαδικό μήκος που φράσσεται από $\mathcal{O}(p^2 + p \tau_f)$ και $\mathcal{L}(f_{red}) = \mathcal{O}(p + \tau_f)$, όπου f_{red} το χωρίς-τετράγωνα μέρος της f .

Πόρισμα 2.6 (Υπολογισμός προσήμου - SIGN_AT). [2, 8] Δοθέντος ενός πραγματικού αλγεβρικού αριθμού $\alpha \cong (f, [a, b])$, με $\mathcal{L}(a) = \mathcal{L}(b) = \mathcal{O}(p\tau_f)$, και ενός $g \in \mathbb{Z}[x]$, τέτοιου που $\deg(g) = q$, $\mathcal{L}(g) = \tau_g$, υπολογίζουμε το πρόσημο $\text{sign}(g(\alpha))$ σε χρόνο (bit-πολυπλοκότητα) $\tilde{O}_B(pq \max\{\tau_f, \tau_g\} + p \min\{p, q\}^2 \tau_f)$.

Λήμμα 2.7 (Aggregate separation). Δοθέντος $f \in \mathbb{Z}[x]$, το άθροισμα των δυαδικών μηκών όλων διαστημάτων απομόνωσης των πραγματικών ριζών του f είναι $\mathcal{O}(p^2 + p\tau_f)$.

Πόρισμα 2.8 (Intermediate Points). Δοθείσης μιας λίστας με τις πραγματικές ρίζες του f υπό μορφή διαστημάτων απομόνωσης, υπολογίζουμε ρητούς μεταξύ τους σε χρόνο $\tilde{O}_B(p^2 + p\tau_f)$.

2.4 Πολυώνυμα πολλών μεταβλητών

Με πολυώνυμα σε πολλές μεταβλητές χρησιμοποιούμε την τεχνική της δυαδικής κατάτμησης (binary segmentation) [27]. Μια εναλλακτική προσέγγιση βρίσκεται στο [14]. Έστω $f, g \in (\mathbb{Z}[y_1, \dots, y_k])[x]$ με $\mathcal{L}(f), \mathcal{L}(g) \leq \tau$, $\deg_x(f) = p \geq q = \deg_x(g)$, $\deg_{y_i}(f) \leq d_i$ και $\deg_{y_i}(g) \leq d_i$. Έστω ακόμη $d = \prod_{i=1}^k d_i$.

Πρόταση 2.9. [27] Υπολογίζουμε την $\mathbf{SRQ}(f, g)$, οποιοδήποτε πολυώνυμο της $\mathbf{SR}(f, g)$ και την απαλοίφουσα $\text{res}(f, g)$ σε χρόνο $\tilde{O}_B(q(p+q)^{k+1}d\tau)$.

Λήμμα 2.10. Η $\mathbf{SR}(f, g)$ υπολογίζεται σε χρόνο $\tilde{O}_B(q(p+q)^{k+2}d\tau)$.

Θεώρημα 2.11. Αποτιμούμε την $\mathbf{SR}(f, g)$ στο $x = \alpha$, όπου $a \in \mathbb{Q} \cup \{\infty\}$ και $\mathcal{L}(a) = \sigma$, σε χρόνο $\tilde{O}_B(q(p+q)^{k+1}d \max\{\tau, \sigma\})$.

Πόρισμα 2.12. Υπολογίζουμε την $\mathbf{SR}(f, g)$ σε χρόνο $\tilde{O}_B(pq(p+q)^2d\tau)$. Για οποιοδήποτε πολυώνυμο $\mathbf{SR}_j(f, g)$ της $\mathbf{SR}(f, g)$, ισχύει $\deg_x(\mathbf{SR}_j(f, g)) = \mathcal{O}(\max\{p, q\})$, $\deg_y(\mathbf{SR}_j(f, g)) = \mathcal{O}(\max\{p, q\}d)$ και $\mathcal{L}(\mathbf{SR}_j(f, g)) = \mathcal{O}(\max\{p, q\}\tau)$.

Πόρισμα 2.13. Υπολογίζουμε την $\mathbf{SRQ}(f, g)$, οποιοδήποτε πολ/μο της $\mathbf{SR}(f, g)$, και την απαλοίφουσα $\text{res}(f, g)$ σε χρόνο $\tilde{O}_B(pq \max\{p, q\}d\tau)$.

Πόρισμα 2.14. Υπολογίζουμε την $\mathbf{SR}(f, g; a)$, όπου $a \in \mathbb{Q} \cup \{\infty\}$ και $\mathcal{L}(a) = \sigma$, σε χρόνο $\tilde{O}_B(pq \max\{p, q\}d \max\{\tau, \sigma\})$. Για τα πολυώνυμα $\mathbf{SR}_j(f, g; a) \in \mathbb{Z}[y]$, εκτός των f, g , έχουμε $\deg_y(\mathbf{SR}_j(f, g; a)) = \mathcal{O}((p+q)d)$ και $\mathcal{L}(\mathbf{SR}_j(f, g; a)) = \mathcal{O}(\max\{p, q\}\tau + \min\{p, q\}\sigma)$.

2.4.1 Υπολογισμός προσήμου πολυων. δύο μεταβλητών

Ανάγουμε τον υπολογισμό του προσήμου του $f \in \mathbb{Z}[x, y]$ πάνω στο $(\alpha, \beta) \in \mathbb{R}_{alg}^2$ σε υπολογισμό προσήμου πάνω σε πολλά σημεία στο \mathbb{Q}^2 . Έστω $\deg_x(f) = \deg_y(f) = n_1$, $\mathcal{L}(f) = \sigma$ και $\alpha \cong (A, [a_1, a_2])$, $\beta \cong (B, [b_1, b_2])$, όπου $A, B \in \mathbb{Z}[X]$, $\deg(A) = \deg(B) = n_2$, $\mathcal{L}(A) = \mathcal{L}(B) = \sigma$. Η ιδέα είναι πως υπολογίζουμε την ακολουθία $\mathbf{SR}(A, f)$ ως προς x , και δημιουργούμε δύο αντίγραφα. Αποτιμούμε το ένα αντίγραφο στο αριστερό άκρο a_1 και το άλλο αντίγραφο στο δεξί άκρο a_2 , και κάθε μια ακολουθία αποτιμάται πάνω στο β με το πόρ. 2.6. Τελικά καταμετρώντας τις εναλλαγές προσήμου μπορούμε να αποφανθούμε για το πρόσημο της f πάνω στο (α, β) . Έτσι ο αλγόριθμος γενικεύει την περίπτωση μιας μεταβλητής, [29, 8, 35] (πόρ. 2.6). Για τα A και B , υποθέτουμε πως ξέρουμε τις τιμές τους στα a_1, a_2 και b_1, b_2 αντίστοιχα.

Θεώρημα 2.15 (BIVARIATE-SIGN_AT). Έστω $f \in \mathbb{Z}[x, y]$ τέτοιο που $\deg_x(f) = \deg_y(f) = n_1$ και $\mathcal{L}(f) = \sigma$ και δύο πραγμ. αλγ. αριθμοί $\alpha \cong (A, \mathcal{J}_\alpha) = [a_1, a_2]$, $\beta \cong (B, \mathcal{J}_\beta) = [b_1, b_2]$ όπου $A, B \in \mathbb{Z}[X]$, $\deg(A) = \deg(B) = n_2$, $\mathcal{L}(A) = \mathcal{L}(B) = \sigma$ και $\mathcal{J}_\alpha, \mathcal{J}_\beta \in \mathbb{Q}^2$. Τότε αποτιμούμε το πρόσημο του f πάνω στα α και β με πολυπλοκότητα $\tilde{O}_B(n_1^2 n_2^3 \sigma)$, υποθέτοντας $n_1 \leq n_2$.

3 Αλγόριθμοι

Εδώ παρουσιάζουμε τις μεθόδους μας για επίλυση 2×2 συστημάτων.

3.1 Αλγόριθμος GRID

Η μέθοδος GRID είναι η άμεση προσέγγιση, δείτε επίσης [9]. Υπολογίζουμε τις πραγματικές λύσεις ως προς x και ως προς y των απαλοιφουσών $\text{res}_x(f, g)$ και $\text{res}_y(f, g)$. Εν συνεχεία, τις ταιριάζουμε με τον αλγόριθμο BIVARIATE-SIGN_AT (θεώρ. 2.15) εξετάζοντας όλα τα παραλληλόγραμμα στο επαγόμενο πλέγμα. Η έξοδος είναι μια λίστα ζευγών πραγματικών αλγεβρικών αριθμών, η οποία αναπαρίσταται υπό μορφή διαστημάτων απομόνωσης. Τα άκρα ορίζουν παραλληλόγραμμα με μοναδική λύση στο εσωτερικό τους.

Η μέθοδος είναι ελκυστική μιας και είναι απλή, αλλά ο υπολογισμός προσήμου είναι πολύ ακριβός. Ο αλγόριθμος δεν απαιτεί γενική θέση: στη συνέχεια παρουσιάζουμε μια γενική μέθοδο στρέβλωσης η οποία φέρνει το σύστημα σε γενική θέση προκειμένου να υπολογιστούν και οι πολλαπλότητες στον ίδιο ασυμπτωτικό χρόνο. Ο αλγόριθμος επιτρέπει τη χρήση ευρετικών όπως είναι ο μικτός όγκος, ή η απαρίθμηση των ριζών σε συγκεκριμένη τετμημένη (βλ. ενότητα 4).

Θεώρημα 3.1. Η απομόνωση όλων των πραγματικών ριζών του συστήματος $f = g = 0$ χρησιμοποιώντας τη μέθοδο GRID έχει πολυπλοκότητα $\tilde{O}_B(n^{14} + n^{13}\sigma)$, δεδομένου $\sigma = \mathcal{O}(n^3)$ ή $\tilde{O}_B(N^{14})$, όπου $N = \max\{n, \sigma\}$.

Στη συνέχεια εξετάζουμε την πολλαπλότητα μιας ρίζας (α, β) του συστήματος. Παρόμοιες εργασίες περιλαμβάνουν [10, 30, 34]. Η μέθοδος μας ανάγεται σε διμετάβλητο υπολογισμό προσήμου και δεν απαιτεί παραγοντοποίηση.

3.1.1 Ντετερμινιστική στρέβλωση (Deterministic shear)

Βρίσκουμε μια επαρκή (οριζόντια) στρέβλωση τέτοια που η εξίσωση

$$R_t(x) = \text{res}_y(f(x + ty, y), g(x + ty, y)), \quad (1)$$

έχει απλές ρίζες και οι οποίες αντιστοιχούν στις προβολές των λύσεων του συστήματος $f(x, y) = g(x, y) = 0$, όπου $t \mapsto t_0 \in \mathbb{Z}$, και ο βαθμός των πολυωνύμων παραμένει ο ίδιος. Για μια διαφορετική προσέγγιση δείτε [12, 2].

Λήμμα 3.2. Ο υπολογισμός ενός $t_0 \in \mathbb{Z}$, τέτοιου που η αντίστοιχη στρέβλωση να είναι επαρκώς γενική, έχει πολυπλοκότητα $\tilde{O}_B(n^{10} + n^9\sigma)$.

Θεώρημα 3.3. Υπό τις προϋποθέσεις του θεωρήματος 3.1, έχοντας απομονώσει όλες τις πραγματικές ρίζες του $f = g = 0$, είναι πιθανό να προσδιορίσουμε τις πολλαπλότητές τους σε χρόνο $\tilde{O}_B(n^{12} + n^{11}\sigma + n^{10}\sigma^2)$.

3.2 Ο αλγόριθμος M_RUR

Η μέθοδος M_RUR υποθέτει πως τα πολυώνυμα είναι σε Γενική Θέση: διαφορετικές ρίζες προβάλλονται σε διαφορετικές τετμημένες και οι συντελεστές ως προς y δεν έχουν κοινές ρίζες.

Πρόταση 3.4. [10, 2] Έστω f, g πολυώνυμα πρώτα μεταξύ τους, σε γενική θέση. Αν $\mathbf{SR}_j(x, y) = \mathbf{sr}_j(x)y^j + \mathbf{sr}_{j,j-1}(x)y^{j-1} + \dots + \mathbf{sr}_{j,0}(x)$, και (α, β) είναι μια πραγματική λύση του συστήματος $f = g = 0$, τότε υπάρχει k , τέτοιο που $\mathbf{sr}_0(\alpha) = \dots = \mathbf{sr}_{k-1}(\alpha) = 0$, $\mathbf{sr}_k(\alpha) \neq 0$ και $\beta = -\frac{1}{k} \frac{\mathbf{sr}_{k,k-1}(\alpha)}{\mathbf{sr}_k(\alpha)}$.

Αυτό εκφράζει την τεταγμένη μιας λύσης με ρητή αναπαράσταση πολυωνύμου μιας μεταβλητής (Rational Univariate Representation (RUR)) ως προς την τετμημένη. Η RUR εφαρμόζεται σε αλγεβρικά συστήματα πολλών μεταβλητών [4, 28, 2] και γενικεύει τη μέθοδο του Kronecker. Ο αλγόριθμός είναι παρόμοιος με τους [12, 10]. Τροποποιήσαμε τον αλγόριθμο [9], ώστε η έξοδος να περιλαμβάνει διαστήματα απομόνωσης, εξού και το όνομα τροποποιημένος-RUR (modified RUR - M_RUR). Η πιο σημαντική διαφορά με το [10] είναι πως εκείνοι χρησιμοποιούν αναπαράσταση Thom για τους πραγματικούς αλγεβρικούς αριθμούς. Προβάλλουμε στους άξονες x και y και για κάθε πραγματική λύση στον άξονα x υπολογίζουμε την τεταγμένη χρησιμοποιώντας την πρότ. 3.4. Αρχικά υπολογίζουμε την ακολουθία $\mathbf{SR}(f, g)$ ως προς y σε χρόνο $\tilde{O}_B(n^5 \sigma)$ (πόρ. 2.12).

Η πρώτη φάση (προβολή) είναι παρόμοια με τον GRID. Η πολυπλοκότητα καθορίζεται από την επίλυση στους πραγματικούς των απαλοιφουσών, δηλαδή $\tilde{O}_B(n^{12} + n^{10} \sigma^2)$. Έστω α_i , αντίστοιχα β_j , να είναι οι πραγματικές συντεταγμένες. Υπολογίζουμε τους ρητούς q_j μεταξύ των β_j σε χρόνο $\tilde{O}_B(n^5 \sigma)$, μέσω της συνάρτησης INTERMEDIATE_POINTS(P_y):

$$q_0 < \beta_1 < q_1 < \beta_2 < \dots < \beta_{\ell-1} < q_{\ell-1} < \beta_\ell < q_\ell, \quad (2)$$

όπου $\ell \leq 2n^2$. Κάθε β_j αντιστοιχεί σε ένα μοναδικό α_i . Η πολλαπλότητα του α_i είναι η πολ/τα μιας πραγμ. λύσης του συστήματος που το έχει σαν τετμημένη. Εν συνεχεία, υπολογίζουμε σε χρόνο $\tilde{O}_B(n^9 + n^8 \sigma)$ ένα k τέτοιο που η πρόταση 3.4 να ικανοποιείται, βλ. [24, 10]. Τέλος, εκμεταλλευόμαστε τη γενική θέση και την εξίσωση (2), ώστε να ταιριάζουμε τις πραγματικές λύσεις του R_x με αυτές του R_y , σε συνολικό χρόνο (όλες οι επαναλήψεις) $\tilde{O}_B(n^{10} + n^9 \sigma)$.

Θεώρημα 3.5. Απομονώνουμε όλες τις πραγματικές ρίζες του συστήματος $f = g = 0$, αν τα f, g είναι σε γενική θέση, με τον αλγόριθμο M_RUR σε χρόνο $\tilde{O}_B(n^{12} + n^{10} \sigma^2)$ · η απλά $\tilde{O}_B(N^{12})$, όπου $N = \max\{n, \sigma\}$.

Η υπόθεση γενικής θέσης είναι χωρίς βλάβη της γενικότητας αφού μπορούμε πάντα να τοποθετήσουμε το σύστημα σε τέτοια θέση εφαρμόζοντας στρέβλωση· δείτε ενότητα 3.1.1 και επίσης [2, 10]. Το δυαδικό μήκος των συντελεστών των πολυωνύμων του (στρεβλωμένου) συστήματος γίνεται $\tilde{O}(n + \sigma)$ [10] και δεν αλλάζει το φράγμα του θεωρ. 3.5. Απομένει να εκφραστούν οι πραγματικές λύσεις στο αρχικό σύστημα συντεταγμένων· κάτι μη τετριμμένο στην πράξη.

3.3 Ο αλγόριθμος G_RUR

Σε αυτή την ενότητα παρουσιάζουμε έναν αλγόριθμο που χρησιμοποιεί μερικές ιδέες από τον M_RUR και στηρίζεται σε υπολογισμούς MKΔ πολυωνύμων με συντελεστές σε σώμα επέκτασης προκειμένου να είναι αποδοτικός (εξού και το όνομα

G_RUR από το gcd). Για αυτούς τους υπολογισμούς MKΔ χρησιμοποιούμε τον αλγόριθμο και την υλοποίηση σε MAPLE των [32].

Τα πρώτα βήματα είναι παρόμοια με αυτά των προηγούμενων αλγορίθμων: Προβάλλουμε στους άξονες, επιλύουμε στους πραγματικούς και υπολογίζουμε τα ενδιάμεσα σημεία στον άξονα y . Η μέθοδος έχει πολυπλοκότητα $\tilde{O}_B(n^{12} + n^{10}\sigma^2)$. Για κάθε τετμημένη x , έστω α , υπολογίζουμε το χωρίς τετράγωνα μέρος των $f(\alpha, y)$ και $g(\alpha, y)$, έστω \bar{f} και \bar{g} . Υποθέτοντας ταχείς αλγορίθμους πολλαπλασιασμού, κάτι το οποίο στο [32] δεν είχε γίνει, επαναλαμβάνουμε αυτή τη διαδικασία $O(n^2)$ φορές, με αποτέλεσμα το συνολικό κόστος να είναι $\tilde{O}_B(n^6\sigma^2)$. Τώρα για κάθε α , υπολογίζουμε το $h = \gcd(\bar{f}, \bar{g})$ με συνολικό κόστος $\tilde{O}_B(n^8 + n^6\sigma^2)$. Οι πραγματικές λύσεις του h αντιστοιχούν στις λύσεις του συστήματος με τετμημένη α . Η καρδιά της μεθόδου βρίσκεται στο γεγονός πως το h αλλάζει πρόσημο μόνο πάνω στο διάστημα το οποίο περιέχει τις πραγματικές του ρίζες. Προκειμένου να ελέγξουμε αυτά τα πρόσημα, αρκεί να αντικαταστήσουμε το y στο h από τα ενδιάμεσα σημεία, παίρνοντας έτσι ένα πολυώνυμο στο $\mathbb{Z}[\alpha]$, βαθμού $O(n)$. Τώρα κοιτάμε αυτό το πολυώνυμο στον $\mathbb{Z}[x]$ και το αποτιμούμε πάνω στο α με κόστος $\tilde{O}_B(n^6 + n^5\sigma + n^4s_j)$. Αθροίζοντας πάνω σε $O(n^2)$ σημεία και από το λήμμα 2.7 παίρνουμε $\tilde{O}_B(n^8 + n^7\sigma)$, με συνολική πολυπλοκότητα $\tilde{O}_B(n^{10} + n^9\sigma)$.

Θεώρημα 3.6. Απομονώνουμε τις πραγματικές λύσεις του συστήματος $f = g = 0$, με τη μέθοδο G_RUR σε χρόνο $\tilde{O}_B(n^{12} + n^{10}\sigma^2)$ ή $\tilde{O}_B(N^{12})$, όπου $N = \max\{n, \sigma\}$.

4 Εφαρμογές, Υλοποίηση και Πειράματα

Οι μέθοδοι που παρουσιάσαμε βρίσκουν εφαρμογή στην απαρίθμηση πραγμ. ριζών πολυωνύμων με συντελεστές σε σώμα επέκτασης, σε πολλές ανισώσεις με ακέραιους συντελεστές σε δύο μεταβλητές, καθώς επίσης και στην πολυπλοκότητα υπολογισμού τοπολογίας αλγεβρικής καμπύλης στον \mathbb{R}^2 . Για περισσότερα δείτε [6, 5]. Στη συνέχεια παρουσιάζουμε την υλοποίησή μας ανοιχτού κώδικα σε MAPLE¹ και αναδεικνύει τις δυνατότητές του συγκριτικά με άλλα πακέτα. Παρέχουμε μεθόδους για προσημασμένες ακολουθίες υπολοίπων, επίλυση στους πραγματικούς μέσω του αλγορίθμου του Sturm, υπολογισμούς με έναν και δύο πραγματικούς αλγεβρικούς αριθμούς, όπως ο υπολογισμός προσήμου και η σύγκριση, και επίλυση 2×2 συστημάτων με ακέραιους συντελεστές.

4.1 Οι αλγόριθμοί μας

Εξετάσαμε τα πολυωνυμικά συστήματα που παρουσιάζονται στα [6, 5]. Τα συστήματα R_i, M_i, D_i είναι από το [9], τα C_i από [12] και τα $W_i, i = 1, \dots, 4$, είναι τα C_i αφού εναλλάξουμε τους ρόλους των x, y . Για υπολογισμούς MKΔ σε σώμα επέκτασης, χρησιμοποιήσαμε το [32]. Οι βέλτιστοι αλγόριθμοι για τον υπολογισμό και την αποτίμηση ακολουθιών υπολοίπων δεν έχουν υλοποιηθεί ακόμη.

Τα κύρια πειραματικά αποτελέσματά μας φαίνονται στον πίνακα 1 κάτω από τον τίτλο SLV (Sturm soLVer). Η απόδοση όλων των υλοποιήσεων είναι ο μέσος όρος 10 εκτελέσεων σε MAPLE 9.5 σε γραμμή εντολών σε έναν 2GHz AMD64@3K+ επεξεργαστή με 1GB RAM. Ο G_RUR είναι επικρατέστερος αφού είναι γρηγορότερος από τον GRID και τον M_RUR σε 17 από τις 18 περιπτώσεις. Ίσως αυτό να μην

¹www.di.uoa.gr/~erga/soft/SLV_index.html

ισχύει όταν ο βαθμός στο σώμα επέκτασης είναι μεγάλος. Όλοι οι αλγόριθμοι χρησιμοποιούν φίλτρα προκειμένου να είναι αποδοτικοί. Το κυριότερο από αυτά είναι η αριθμητική διαστημάτων προκειμένου να αποφεύγουμε τις χρονοβόρες αποτιμήσεις ακολουθιών πολυωνύμων όπου αυτό είναι δυνατό. Για περισσότερες πληροφορίες δείτε το [6]. Αναλύοντας τους χρόνους παρατηρούμε σε γενικές γραμμές: ο GRID ξοδεύει περίπου το 73% του χρόνου εκτέλεσης στο ταίριασμα λύσεων, ο M_RUR το 45-50% του χρόνου εκτέλεσης στο ταίριασμα και ένα 24-27% στα φίλτρα και τέλος ο G_RUR το 55-80% του χρόνου στο ταίριασμα, συμπεριλαμβάνοντας το χρόνο για υπολογισμούς MKΔ σε σώμα επέκτασης. Τα ποσοστά αυτά είναι ελαφρά αυξημένα όταν ο GRID και ο G_RUR επιλύουν στρεβλωμένα συστήματα.

4.2 Άλλες υλοποιήσεις

Το FGB/RS² [28] πραγματοποιεί επίλυση στους πραγματικούς χρησιμοποιώντας βάσεις Gröbner και RUR, μέσω του περιβάλλοντος που παρέχει σε MAPLE· επιπλέον ρυθμίσεις μπορούν να βελτιώσουν το χρόνο απόκρισης κατά 20-30%. Ελέγξαμε επίσης τρία πακέτα της SYNAPS³: ο STURM είναι μια απλοϊκή έκδοση του GRID [9]· ο SUBDIV υλοποιεί το [23], χρησιμοποιώντας βάση Bernstein και αριθμητική διπλής ακριβείας. Απαιτεί ένα αρχικό πλαίσιο και το $[-10, 10] \times [-10, 10]$ χρησιμοποιήθηκε. Ο NEWMAC [25] είναι γενικός και βασίζεται σε ιδιοδιανύσματα με χρήση του πακέτου LAPACK και υπολογίζει όλες τις μιγαδικές λύσεις.

MAPLE υλοποιήσεις: Ο INSULATE υλοποιεί το [34] για τον υπολογισμό της τοπολογίας μιας πραγματικής αλγεβρικής καμπύλης και ο TOP υλοποιεί το [12]. Οι δύο αυτές υλοποιήσεις μας παραχωρήθηκαν από τους αντίστοιχους συγγραφείς. Προσπαθήσαμε να τις τροποποιήσουμε ώστε να τερματίζουν μόλις υπολογίζουν τις πραγματικές λύσεις που αντιστοιχούν στο επαγόμενο διμετάβλητο σύστημα. Δεν ήταν όμως εύκολο να κάνουμε τέτοιες τροποποιήσεις ώστε να αντιμετωπίζουν γενικά συστήματα και για το λόγο αυτό δεν υπάρχουν χρόνοι στο πρώτο σετ πειραμάτων. Ο TOP έχει μια παράμετρο που καθορίζει την αρχική ακρίβεια (δεκαδικά ψηφία)· δεν υπάρχει εύκολος τρόπος να επιλέξει κανείς μια καλή αρχική τιμή. Έτσι, ακολουθήσαμε το [13] και καταγράψαμε την απόδοσή του για ακρίβεια 60 και 500 ψηφίων.

Συγκεντρωτικά αποτελέσματα εμφανίζονται στον πίνακα 1. Στις περιπτώσεις που κάποια υλοποίηση δεν κατάφερε να βρει σωστό πλήθος πραγματικών λύσεων σηματοδοτούμε με ένα *. Σημειώστε πως ο NEWMAC χρειάζεται ένα ακόμη βήμα ώστε να διαχωρίσει κανείς τις πραγματικές λύσεις μεταξύ όλων των μιγαδικών.

Ο G_RUR είναι γρηγορότερος από το FGB/RS σε 8 από τις 18 περιπτώσεις, συμπεριλαμβάνοντας το C_5 . Είναι επίσης γρηγορότερος από τον STURM σε 6 από τα 18 πειράματα. Συγκριτικά με τον SUBDIV ο G_RUR είναι γρηγορότερος στις μισές περιπτώσεις. Παρατηρήστε πως ο SUBDIV συμπεριφέρεται περίεργα στα C_1 και W_1 . Συγκριτικά με τον NEWMAC, ο G_RUR τα πάει καλύτερα στα M_4, D_1 και W_3 και είναι συγκρίσιμος στα R_1 και R_3 . Σε κάποιες περιπτώσεις όμως ο NEWMAC δεν υπολογίζει όλες τις πραγματικές λύσεις. Σχετικά με τις υλοποιήσεις τοπολογίας, ο G_RUR είναι ταχύτερος από τον INSULATE σε όλα τα συστήματα εκτός του W_2 . Συγκριτικά με τον TOP ξεκινώντας με 60 ψηφία, ο G_RUR είναι ταχύτερος σε όλα τα συστήματα πλην του W_2 . Με 500 ψηφία, ο TOP εξακολουθεί να είναι ταχύτερος στο W_2 . Καθώς η διάσταση των πολυωνυμικών συστημάτων

²<http://www-spaces.lip6.fr/index.html>

³<http://www-sop.inria.fr/galaad/logiciels/synaps/>

| (Σ) | βαθμός | | κ λ μ ν | Μέσος Χρόνος (msecs) | | | | | | | | | ΤΟΠΟΛΟΓΙΑ | | |
|----------------|----------------|----|------------------|---------------------------|--------|-------|-------|-------------------|----------|--------|---------|--------|-----------|-----|---|
| | | | | ΕΠΙΛΥΣΗ ΣΕ ΔΥΟ ΜΕΤΑΒΛΗΤΕΣ | | | | | | SYNAPS | | | INSULATE | TOP | |
| | f | g | | GRID | M_RUR | G_RUR | FG/RS | STURM | SUBDIV | NEWMAC | 60 | 500 | | | |
| | R ₁ | 3 | | 4 | 2 | 5 | 9 | 5 | 26 | 2 | 2 | 5 | – | – | – |
| R ₂ | 3 | 1 | 1 | 66 | 21 | 36 | 24 | 1 | 1 | 1 | – | – | – | | |
| R ₃ | 3 | 1 | 1 | 1 | 2 | 1 | 22 | 1 | 2 | 1 | – | – | – | | |
| M ₁ | 3 | 3 | 4 | 87 | 72 | 10 | 25 | 2 | 1 | 2 | – | – | – | | |
| M ₂ | 4 | 2 | 3 | 4 | 5 | 4 | 24 | 1 | 289* | 2 | – | – | – | | |
| M ₃ | 6 | 3 | 5 | 803 | 782 | 110 | 30 | 230 | 5,058* | 7 | – | – | – | | |
| M ₄ | 9 | 10 | 2 | 218 | 389 | 210 | 158 | 90 | 3* | 447 | – | – | – | | |
| D ₁ | 4 | 5 | 1 | 6 | 12 | 6 | 28 | 2 | 5 | 8 | – | – | – | | |
| D ₂ | 2 | 2 | 4 | 667 | 147 | 128 | 26 | 21 | 1* | 2 | – | – | – | | |
| C ₁ | 7 | 6 | 6 | 1,896 | 954 | 222 | 93 | 479 | 170,265* | 39 | 524 | 409 | 1,367 | | |
| C ₂ | 4 | 3 | 6 | 177 | 234 | 18 | 27 | 12 | 23* | 4 | 28 | 36 | 115 | | |
| C ₃ | 8 | 7 | 13 | 580 | 1,815 | 75 | 54 | 23 | 214* | 25 | 327 | 693 | 2,829 | | |
| C ₄ | 8 | 7 | 17 | 5,903 | 80,650 | 370 | 138 | 3,495 | 217* | 190* | 1,589 | 1,624 | 6,435 | | |
| C ₅ | 16 | 15 | 17 | > 20 ^l | 60,832 | 3,877 | 4,044 | > 20 ^l | 6,345* | 346* | 179,182 | 91,993 | 180,917 | | |
| W ₁ | 7 | 6 | 9 | 2,293 | 2,115 | 247 | 92 | 954 | 55,040* | 39 | 517 | 419 | 1,350 | | |
| W ₂ | 4 | 3 | 5 | 367 | 283 | 114 | 29 | 20 | 224* | 3 | 27 | 20 | 60 | | |
| W ₃ | 8 | 7 | 13 | 518 | 2,333 | 24 | 56 | 32 | 285* | 25 | 309 | 525 | 1,588 | | |
| W ₄ | 8 | 7 | 17 | 5,410 | 77,207 | 280 | 148 | 4,086 | 280* | 207* | 1,579 | 1,458 | 4,830 | | |

Πίνακας 1: Απόδοση του λογισμικού μας και άλλων υλοποιήσεων.

αυξάνει, ο G_RUR φαίνεται να είναι πιο αποτελεσματικός και από τις δύο αυτές υλοποιήσεις.

Αναφορές

- [1] D. Arnon and S. McCallum. A polynomial time algorithm for the topological type of a real algebraic curve. *JSC*, 5:213–236, 1988.
- [2] S. Basu, R. Pollack, and M-F. Roy. *Algorithms in Real Algebraic Geometry*, volume 10 of *Algorithms and Computation in Mathematics*. Springer-Verlag, 2nd edition, 2006.
- [3] J. Canny. *The Complexity of Robot Motion Planning*. ACM – MIT Press Doctoral Dissertation Award Series. MIT Press, Cambridge, MA, 1987.
- [4] J. Canny. Some algebraic and geometric computations in PSPACE. In *STOC*, 460–467, 1988.
- [5] D.I. Diochnos, I.Z. Emiris, and E.P. Tsigaridas. On the Complexity of Real Solving Bivariate Systems. In *ISSAC*, 2007.
- [6] D.I. Diochnos, I.Z. Emiris, and E.P. Tsigaridas. On the asymptotic and practical complexity of solving bivariate systems over the reals. *Journal of Symbolic Computation*, 2008.
- [7] A. Eigenwillig, V. Sharma, and C.K. Yap. Almost tight recursion tree bounds for the Descartes method. In *ISSAC '06: Proceedings of the 2006 International Symposium on Symbolic and Algebraic Computation*, 71–78, New York, NY, USA, 2006. ACM Press.
- [8] I.Z. Emiris, B. Mourrain, and E.P. Tsigaridas. Real Algebraic Numbers: Complexity Analysis and Experimentation. In Hertling, P. and Hoffmann,

C. and Luther, W. and Revol, N., editors, *Reliable Implementations of Real Number Algorithms: Theory and Practice*, volume 5045 of *LNCS*, 57–82. Springer Verlag, 2008. also available in www.inria.fr/rrrt/rr-5897.html.

- [9] I.Z. Emiris and E.P. Tsigaridas. Real solving of bivariate polynomial systems. In V. Ganzha and E. Mayr, editor, *Proc. Computer Algebra in Scientific Computing (CASC)*, vol. 3718 of *LNCS*, 150–161. Springer, 2005.
- [10] L. González-Vega and M. El Kahoui. An Improved Upper Complexity Bound for the Topology Computation of a Real Algebraic Plane Curve. *J. Complexity*, 12(4):527–544, 1996.
- [11] L. González-Vega, H. Lombardi, T. Recio, and M-F. Roy. Sturm-Habicht Sequence. In *ISSAC*, 136–146, 1989.
- [12] L. González-Vega and I. Necula. Efficient topology determination of implicitly defined algebraic plane curves. *Computer Aided Geometric Design*, 19(9):719–743, Dec 2002.
- [13] M. Kerber. Analysis of Real Algebraic Plane Curves. Diploma thesis, MPI Saarbrücken, 2006.
- [14] J. Klose. Binary Segmentation for Multivariate Polynomials. *J. Complexity*, 11(3):330–343, 1995.
- [15] K.H. Ko, N.M. Patrikalakis, and T. Sakkalis. Resolution of multiple roots of nonlinear polynomial systems. *International J. of Shape Modeling*, 11(1):121–147, 2005.
- [16] D. Lakshman, Y.N. and Lazard. On the complexity of zero-dimensional algebraic systems. In T.Mora and C. Traverso, editors, *Effective Methods in Algebraic Geometry*, volume 94 of *Progress in Mathematics*, 217–225, Boston, 1991. Birkhäuser. (Proc. MEGA '90, Livorno, Italy).
- [17] T. Lickteig and M-F. Roy. Sylvester-habicht sequences and fast Cauchy index computation. *J. Symb. Comput.*, 31(3):315–341, 2001.
- [18] H. Lombardi, M-F. Roy, and M. Safey El Din. New Structure Theorem for Subresultants. *J. Symb. Comput.*, 29(4-5):663–689, 2000.
- [19] P.S. Milne. On the solution of a set of polynomial equations. In B. Donald, D. Kapur, and J. Mundy, editors, *Symbolic and Numerical Computation for Artificial Intelligence*, 89–102. Academic Press, 1992.
- [20] B. Mourrain. A new criterion for normal form algorithms. *Proc. AAECC*, vol. 1719 of *LNCS*, 430–443, 1999.
- [21] B. Mourrain and V.Y. Pan. Solving special polynomial systems by using structured matrices and algebraic residues. In F. Cucker and M. Shub, editors, *Proc. Workshop on Foundations of Computational Mathematics*, 287–304, Berlin, 1997. Springer-Verlag.
- [22] B. Mourrain and V.Y. Pan. Asymptotic acceleration of solving polynomial systems. In *STOC*, 488–496. ACM Press, New York, 1998.

- [23] B. Mourrain and J-P. Pavone. Subdivision methods for solving polynomial equations. Technical Report RR-5658, INRIA Sophia-Antipolis, 2005.
- [24] B. Mourrain, S. Pion, S. Schmitt, J-P. T  court, E. Tsigaridas, and N. Wolpert. Algebraic issues in computational geometry. In Jean-Daniel Boissonnat and Monique Teillaud, editors, *Effective Computational Geometry for Curves and Surfaces*, 117–155. Springer-Verlag, Mathematics and Visualization, 2006.
- [25] B. Mourrain and Ph. Tr  buchet. Solving projective complete intersection faster. In *ISSAC*, 231–238. ACM Press, New York, 2000.
- [26] P. Pedersen, M-F. Roy, and A. Szpirglas. Counting real zeros in the multivariate case. In F.   yssette and A. Galligo, editors, *Computational Algebraic Geometry*, vol. 109 of *Progress in Mathematics*, 203–224. Birkh  user, Boston, 1993.
- [27] D. Reischert. Asymptotically Fast Computation of Subresultants. In *ISSAC*, 233–240, 1997.
- [28] F. Rouillier. Solving zero-dimensional systems through the rational univariate representation. *Journal of Applicable Algebra in Engineering, Communication and Computing*, 9(5):433–461, 1999.
- [29] T. Sakkalis. Signs of algebraic numbers. *Computers and Mathematics*, 131–134, 1989.
- [30] T. Sakkalis and R. Farouki. Singular Points of Algebraic Curves. *J. Symb. Comput.*, 9(4):405–421, 1990.
- [31] Elias P. Tsigaridas. *Algebraic Algorithms and Applications to Geometry*. PhD thesis, Dept. of Informatics and Telecommunications, University of Athens, 2006.
- [32] M. van Hoeij and M. Monagan. A modular GCD algorithm over number fields presented with multiple extensions. In *ISSAC*, 109–116, July 2002.
- [33] J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge Univ. Press, Cambridge, U.K., 2nd edition, 2003.
- [34] N. Wolpert and R. Seidel. On the Exact Computation of the Topology of Real Algebraic Curves. In *SoCG*. ACM, 2005.
- [35] C.K. Yap. *Fundamental Problems of Algorithmic Algebra*. Oxford University Press, New York, 2000.

Προηγμένες Μέθοδοι Αναζήτησης για Προβλήματα Ικανοποίησης Περιορισμών

Φοίβος Θεοχάρης*

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών
Τμήμα Πληροφορικής και Τηλεπικοινωνιών
Πανεπιστημιούπολη, 157 84 Αθήνα
f.theoharis@di.uoa.gr

Περίληψη Τα τελευταία χρόνια στο πεδίο της Ικανοποίησης Περιορισμών έχει αναπτυχθεί μία πληθώρα προηγμένων μεθόδων αναζήτησης οι οποίες εκμεταλλεύονται καινοτόμες ιδέες στην προσπάθειά τους να λύσουν ολοένα και δυσκολότερα προβλήματα. Η πτυχιακή εργασία αυτή συγκεντρώνει τις σημαντικότερες από αυτές σε μία αντικειμενοστραφή βιβλιοθήκη η οποία μπορεί να χρησιμοποιηθεί σε συνδυασμό με τον επιλυτή NaXos [20] για την εύκολη διατύπωση και ακόμη αποδοτικότερη επίλυση προβλημάτων ικανοποίησης περιορισμών. Επιπροσθέτως προτείνονται και δύο νέες μέθοδοι, η απόδοση των οποίων φαίνεται να είναι εφάμιλλη με των υπολοίπων στην επίλυση αρκετών προβλημάτων.

Λέξεις-κλειδιά: προγραμματισμός με περιορισμούς, μη πλήρης αναζήτηση, ασυμφωνία

1 Εισαγωγή

Τι κοινό έχουν μία αεροπορική εταιρία, ένα περιοδικό με σταυρόλεξα και sudoku και το ωρολόγιο πρόγραμμα μαθημάτων μίας πανεπιστημιακής σχολής; Η απάντηση είναι ότι αξιοποιούν κάποιες από τις πάμπολες εφαρμογές του πεδίου της Τεχνητής Νοημοσύνης που ονομάζεται Ικανοποίηση Περιορισμών. (Constraint Satisfaction). Η χρονοδρομολόγηση προσωπικού των αεροπορικών εταιριών, η παραγωγή και επίλυση σταυρολέξων και γρίφων, η δημιουργία ωρολόγιου προγράμματος είναι λίγα μόνο από τα προβλήματα που λύνονται στην πράξη με τη χρήση τεχνικών που αναπτύχθηκαν στο πεδίο αυτό.

Η συνεισφορά της εργασίας αυτής είναι πρακτική και θεωρητική ταυτόχρονα. Πρακτική γιατί δημιουργήθηκε μία αντικειμενοστραφής βιβλιοθήκη με μεθόδους αναζήτησης για την επίλυση προβλημάτων ικανοποίησης περιορισμών. Θεωρητική γιατί αναπτύχθηκαν δύο καινούριες, ελπιδοφόρες μέθοδοι αναζήτησης και συμπεριελήφθησαν στη βιβλιοθήκη αυτή. Διάφορα προβλήματα χρησιμοποιήθηκαν για την πειραματική αξιολόγηση της κάθε μεθόδου και για τη δοκιμή της βιβλιοθήκης στην πράξη.

* Επιβλέπων Καθηγητής: Παναγιώτης Σταματόπουλος

2 Προβλήματα Ικανοποίησης Περιορισμών

Τυπικά, ένα πρόβλημα ικανοποίησης περιορισμών (constraint satisfaction problem – CSP) ορίζεται από [13]:

- ένα σύνολο μεταβλητών $X = \{X_1, X_2, \dots, X_n\}$
- για κάθε μεταβλητή X_i ένα μη κενό πεδίο (domain) D_i των δυνατών τιμών της
- ένα σύνολο περιορισμών $C = \{C_1, C_2, \dots, C_m\}$. Κάθε περιορισμός δρα σε ένα σύνολο $S \subseteq X$ και καθορίζει τους επιτρεπτούς συνδυασμούς τιμών που μπορούν να πάρουν οι μεταβλητές που ανήκουν στο S .

Τα πεδία D_i στα πλαίσια αυτής της πτυχιακής θα είναι πεπερασμένα σύνολα αποτελούμενα από διακριτές τιμές. Μία Κατάσταση (state) του προβλήματος ορίζεται με ανάθεση τιμών σε κάποιες από τις μεταβλητές, $\{X_i = v_i, X_j = v_j, \dots\}$. Αν μία ανάθεση δεν παραβιάζει κανέναν από τους περιορισμούς, τότε ονομάζεται συνεπής (consistent). Μία ανάθεση που περιλαμβάνει όλες τις μεταβλητές ονομάζεται πλήρης. Λύση του προβλήματος ικανοποίησης περιορισμών είναι μία πλήρης, συνεπής ανάθεση.

Το πρόβλημα της εύρεσης μίας λύσης σε ένα πρόβλημα ικανοποίησης περιορισμών είναι γενικά δύσκολο υπολογιστικά. Αρκεί να αναφερθεί ότι NP-πλήρη προβλήματα όπως το 3-SAT, η ΚΛΙΚΑ και ο ΚΥΚΛΟΣ ΧΑΜΙΛΤΟΝ είναι προβλήματα ικανοποίησης περιορισμών.

3 Αναζήτηση στην τεχνητή νοημοσύνη

Ας θεωρήσουμε το παρακάτω πρόβλημα: Τρεις ιεραπόστολοι και τρεις καννίβαλοι βρίσκονται στην όχθη ενός ποταμού και πρέπει να περάσουν απέναντι. Στη διάθεσή τους βρίσκεται μία βάρκα χωρητικότητας δύο ατόμων. Σε κάθε όχθη ανά πάσα στιγμή πρέπει οι καννίβαλοι να μην είναι περισσότεροι από τους ιεραποστόλους, αλλιώς θα τους φάνε! Προφανώς η βάρκα δεν μπορεί να περάσει από μόνη της το ποτάμι, οπότε σε κάθε διαδρομή θα πρέπει να βρίσκεται ένα τουλάχιστον άτομο μέσα. Με ποιο τρόπο θα διασχίσουν το ποτάμι;

Πριν επιχειρήσουμε τη λύση, θα ορίσουμε τυπικά την έννοια πρόβλημα. Τέσσερα είναι τα στοιχεία που ορίζουν ένα πρόβλημα [13]:

- Η αρχική κατάσταση. Στο παράδειγμά μας μπορούμε να αναπαραστήσουμε την αρχική κατάσταση ως 331, όπου το πρώτο ψηφίο αντιστοιχεί στον αριθμό των ιεραποστόλων στην όχθη εκκίνησης, το δεύτερο αντιστοιχεί στον αριθμό των καννιβάλων στην όχθη αυτή και το τρίτο ψηφίο είναι 0 αν η βάρκα βρίσκεται στην αντίπερα όχθη ή 1 αν βρίσκεται στην όχθη εκκίνησης.
- Η συνάρτηση διαδόχων (successor function). Η συνάρτηση αυτή παίρνει ως όρισμα μία κατάσταση x του προβλήματος και επιστρέφει ένα σύνολο διατεταγμένων ζευγών (ενέργεια, διάδοχος). Το ποιες ενέργειες είναι επιτρεπτές εξαρτάται από την τρέχουσα κατάσταση x . Η διάδοχος κατάσταση προκύπτει από την εφαρμογή της ενέργειας στη x . Στο παράδειγμα η Διάσχιση(0, 2), δηλαδή

να διασχίσουν το ποτάμι 0 ιεραπόστολοι και 2 καννίβαλοι, είναι μία ενέργεια που μπορεί να εφαρμοστεί στην αρχική κατάσταση. Η κατάσταση στην οποία οδηγεί είναι η 310. Η συνάρτηση διαδόχων με όρισμα την αρχική κατάσταση θα επέστρεφε το εξής σύνολο:

$$\{ (\text{Διάσχιση}(0, 1), 320), \\ (\text{Διάσχιση}(0, 2), 310), \\ (\text{Διάσχιση}(1, 1), 220) \}$$

Το σύνολο όλων των καταστάσεων στις οποίες μπορούμε να οδηγηθούμε από την αρχική κατάσταση με διαδοχική εφαρμογή της συνάρτησης διαδόχων ονομάζεται *χώρος καταστάσεων* (state space) του προβλήματος. Μπορούμε να τον αναπαραστήσουμε με ένα γράφο, του οποίου οι κόμβοι είναι οι καταστάσεις και οι ακμές καθορίζονται από τη συνάρτηση διαδόχων. Ένα μονοπάτι στο γράφο αυτό θα ονομάζεται *μονοπάτι* (path) και στο χώρο καταστάσεων.

- Ο *έλεγχος στόχου* (goal test), με τον οποίον καθορίζουμε ποια ή ποιες καταστάσεις είναι καταστάσεις στόχου. Στο παράδειγμά μας ο έλεγχος στόχου απλά κοιτάζει αν η δεδομένη κατάσταση είναι η 000 (δηλαδή αν όλοι οι ιεραπόστολοι και όλοι οι καννίβαλοι έχουν περάσει απέναντι).
- Η συνάρτηση *κόστους μονοπατιού*, που είναι το άθροισμα των κοστών των ενεργειών που αποτελούν ένα μονοπάτι. Στο παράδειγμά μας θεωρούμε ότι κάθε ενέργεια έχει κόστος ένα, δηλαδή προτιμάμε το μονοπάτι που οδηγεί στην κατάσταση στόχο με όσο το δυνατόν λιγότερες ενέργειες.

Ένα μονοπάτι που συνδέει την αρχική κατάσταση με μία από τις καταστάσεις στόχου ονομάζεται *Λύση* του προβλήματος. Η λύση με το ελάχιστο κόστος καλείται *βέλτιστη*.

Πώς βρίσκουμε λοιπόν μία λύση σε ένα πρόβλημα; Η απάντηση είναι ότι χρησιμοποιούμε αλγορίθμους αναζήτησης, που εξετάζουν ένα δένδρο αναζήτησης (search tree), το οποίο κατασκευάζεται από την αρχική κατάσταση μέσα από τη συνάρτηση διαδόχων. Γενικά, όπως και στο παράδειγμά μας, έχουμε έναν γράφο αναζήτησης. Σχεφτείτε τι γίνεται αν ένας καννίβαλος πάρει τη βάρκα, περάσει απέναντι μόνος του και επιστρέψει μόνος του. Έχουμε έναν κύκλο που επιστρέφει στην ίδια κατάσταση. Ευτυχώς όμως, όπως θα δούμε και παρακάτω, στα προβλήματα ικανοποίησης περιορισμών δεν έχουμε τέτοιες επιπλοκές, κι έτσι θα αρκεστούμε στα δένδρα αναζήτησης.

Σε κάθε κατάσταση αντιστοιχεί ένας κόμβος του δένδρου αναζήτησης. Η σειρά με την οποία εξετάζονται οι κόμβοι εξαρτάται από τη στρατηγική αναζήτησης που χρησιμοποιούμε.

3.1 Η Αναζήτηση στα προβλήματα ικανοποίησης περιορισμών

Ας επιστρέψουμε στα προβλήματα ικανοποίησης περιορισμών. Πώς θα έμοιαζε ένα δένδρο αναζήτησης σε ένα από αυτά τα προβλήματα; Ανά πάσα στιγμή έχουμε να επιλέξουμε μία μεταβλητή στην οποία θα αναθέσουμε τιμή και την τιμή που θα της αναθέσουμε. Αυτό σημαίνει ότι ο παράγοντας διακλάδωσης είναι τρομακτικά μεγάλος, $(n - 1)d$ στο πρώτο βήμα, όπου n είναι ο αριθμός των μεταβλητών και d ο (μέγιστος) αριθμός των πιθανών τιμών κάθε μεταβλητής. Τα πράγματα όμως δεν

είναι ακριβώς έτσι, μιας και τα προβλήματα ικανοποίησης περιορισμών έχουν την ιδιότητα της αντιμεταθετικότητας. Δηλαδή, με οποιαδήποτε σειρά και αν επιλέξουμε τις μεταβλητές θα καταλήξουμε στις ίδιες αναθέσεις. Επομένως σε κάθε βήμα η επιλογή της μεταβλητής είναι τελεσίδικη και ο παράγοντας διακλάδωσης είναι απλώς d .

Αναζήτηση με οπισθοδρόμηση Μία σημαντική μέθοδος αναζήτησης είναι η *αναζήτηση με οπισθοδρόμηση* (backtracking search) που είναι μια μορφή DFS (αναζήτησης πρώτα κατά βάθος). Κάθε φορά που πρόκειται να κάνει ανάθεση μίας τιμής σε κάποια μεταβλητή ελέγχει αν η τιμή αυτή είναι συνεπής ως προς την τρέχουσα ανάθεση. Αν κάποιος περιορισμός παραβιάζεται δοκιμάζεται η επόμενη τιμή. Αν εξαντληθούν οι τιμές ο αλγόριθμος οπισθοδρομεί και έτσι αποφεύγεται η εξερεύνηση ενός συνήθως μεγάλου κομματιού του χώρου αναζήτησης.

Το κύριο ελάττωμα της μεθόδου είναι ότι δεν ανιχνεύει κάποια ασυνέπεια παραμόνο αφότου αυτή συμβεί. Αυτό γιατί ελέγχει για ύπαρξη τυχόν ασυνεπειών μόνο μεταξύ των δεσμευμένων μεταβλητών και δεν εξετάζει καθόλου τις μη δεσμευμένες μεταβλητές. Για να αποφύγουμε αυτό το πρόβλημα θα χρησιμοποιήσουμε την ιδέα της διάδοσης περιορισμών.

Διατήρηση συνέπειας ακμών Η εφαρμογή ενός αλγορίθμου συνέπειας ακμών δεν είναι αρκετή για να λύσει τα περισσότερα προβλήματα ικανοποίησης περιορισμών. Η μέθοδος αναζήτησης με το όνομα *διατήρηση συνέπειας ακμών* (maintaining arc consistency — MAC) [14] μπορεί να βελτιώσει μεθόδους αναζήτησης όπως η αναζήτηση με οπισθοδρόμηση. Ενώ στην κλασική αναζήτηση με οπισθοδρόμηση ελέγχεται η ικανοποίηση των περιορισμών που αφορούν αποκλειστικά σε δεσμευμένες μεταβλητές, κατά τη διατήρηση συνέπειας ακμών λαμβάνονται υπόψη και οι υπόλοιποι περιορισμοί. Συγκεκριμένα, μετά από κάθε ανάθεση εφαρμόζεται ένας από τους αλγορίθμους συνέπειας ακμών για να αφαιρεθούν από τα πεδία των μεταβλητών τιμές που θα παραβίαζαν τους περιορισμούς. Όλοι οι αλγόριθμοι της βιβλιοθήκης Amorgos στηρίζονται στη φιλοσοφία του MAC.

Ευριστικοί μηχανισμοί Για να αυξήσουμε ακόμη περισσότερο την ισχύ του MAC πρέπει να απαντήσουμε τις εξής δύο ερωτήσεις:

1. Ποια μεταβλητή επιλέγουμε για να της κάνουμε ανάθεση σε κάθε βήμα;
2. Με ποια σειρά δοκιμάζουμε τις τιμές της μεταβλητής αυτής;

Ευριστικοί μηχανισμοί επιλογής μεταβλητής Η απάντηση στην πρώτη ερώτηση είναι οι ευριστικοί μηχανισμοί επιλογής μεταβλητής, που εκμεταλλεύονται την πληροφορία που είναι διαθέσιμη σε κάθε βήμα. Η γενική αρχή για μηχανισμούς αυτού του είδους είναι ότι θέλουμε οι αποτυχίες να συμβαίνουν νωρίς (first-fail). Έτσι κλαδεύεται μεγάλο κομμάτι του δένδρου αναζήτησης. Δύο ευριστικοί μηχανισμοί αυτού του είδους είναι ο μηχανισμός των *ελαχίστων απομεινουσών τιμών* (minimum remaining values — MRV) και ο *ευριστικός μηχανισμός βαθμού* (degree heuristic).

Ευριστικοί μηχανισμοί επιλογής τιμής Ο καθορισμός της σειράς με την οποία θα δοκιμάσουμε τις τιμές σε μία μεταβλητή προφανώς παίζει ρόλο μόνο όταν ψάχνουμε μία μόνο λύση στο πρόβλημα. Αν αναζητούμε όλες τις λύσεις, προφανώς θα αναγκαστούμε να δοκιμάσουμε τελικά όλες τις τιμές. Το ίδιο συμβαίνει και αν δεν υπάρχει λύση. Η γενική αρχή που ακολουθούμε είναι, αντίθετα με πριν, ότι θέλουμε να επιτύχουμε νωρίς (succeed first). Ένας ευριστικός μηχανισμός επιλογής τιμής είναι αυτός της *λιγότερο δεσμευτικής τιμής* (least-constraining-value).

Στα περισσότερα προβλήματα, δυστυχώς, οι ευριστικοί μηχανισμοί επιλογής τιμής γενικής χρήσης είναι ασύμφοροι. Αυτό που κάνουμε λοιπόν είναι, αν το πρόβλημα που θέλουμε να λύσουμε το επιτρέπει, να κατασκευάσουμε έναν ευριστικό μηχανισμό προσαρμοσμένο στο συγκεκριμένο πρόβλημα.

4 Προγραμματισμός με περιορισμούς μέσω του επιλυτή Naxos

Ο επιλυτής προβλημάτων ικανοποίησης περιορισμών Naxos κατασκευάστηκε από τον Ν. Ποθητό στα πλαίσια της δικής του πτυχιακής εργασίας [20]. Πρόκειται για μία βιβλιοθήκη που μπορεί να χρησιμοποιηθεί για να λυθούν προβλήματα μέσω του προγραμματισμού με περιορισμούς. Σαν πλεονεκτήματά του μπορούν να αναφερθούν το ότι είναι εύχρηστος, έχει χαμηλές απαιτήσεις σε μνήμη και έχει απόδοση συγκρίσιμη με ανάλογα συστήματα που κυκλοφορούν στο εμπόριο (Ένα τέτοιο σύστημα, για παράδειγμα, είναι ο ILOG solver).

4.1 Επιβολή συνέπειας και περιορισμοί

Ο Naxos χρησιμοποιεί μία τροποποιημένη έκδοση του αλγορίθμου AC-5 για τη διάδοση των περιορισμών. Επιλέχθηκε αυτός ο αλγόριθμος λόγω της παραμετρικότητάς του και της βέλτιστης συμπεριφοράς όσον αφορά την πολυπλοκότητα. Όταν λέμε παραμετρικότητα εννοούμε ότι κάθε περιορισμός υλοποιείται σαν ξεχωριστή υπορουτίνα και καλείται από τον AC-5 όταν αυτό είναι απαραίτητο.

Έχει υλοποιηθεί μία πληθώρα περιορισμών, τους οποίους ο χρήστης έχει τη δυνατότητα να αξιοποιήσει στις εφαρμογές του. Μπορούν να χρησιμοποιηθούν οι συνήθεις αριθμητικοί, συγκριτικοί και λογικοί τελεστές. Επίσης είναι διαθέσιμοι διάφοροι περιορισμοί για πίνακες, όπως ο AllDifferent (όλες οι μεταβλητές του πίνακα να είναι διαφορετικές μεταξύ τους) και οι εκφράσεις Min, Max και Sum, που συνδέουν το ελάχιστο, μέγιστο στοιχείο ή το άθροισμα των στοιχείων του πίνακα αντίστοιχα με κάποια άλλη έκφραση σε έναν περιορισμό.

4.2 Αναζήτηση λύσης μέσω στόχων

Η αναζήτηση στον Naxos υλοποιείται μέσα από τους στόχους. Η συνήθης λειτουργία ενός στόχου είναι να προκαλεί την ανάθεση μίας τιμής σε μία μεταβλητή ή την αφαίρεση κάποιας τιμής από το πεδίο της. Αν κατά τη διάρκεια της αναζήτησης αφαιρεθούν όλες οι τιμές από το πεδίο κάποιας μεταβλητής τότε ο επιλυτής αναλαμβάνει να ακυρώσει τους στόχους που οδήγησαν στην αποτυχία και να επαναφέρει το γράφο περιορισμών στην προηγούμενη κατάστασή του.

Ένας στόχος, εκτός από την ανάθεση ή την αφαίρεση τιμών από τα πεδία των μεταβλητών, έχει τη δυνατότητα να επιλέξει την επόμενη μεταβλητή στην οποία θα γίνει ανάθεση, ή, κατά τον τερματισμό της εκτέλεσης του, να δημιουργήσει έναν άλλο στόχο. Αυτό το χαρακτηριστικό μας προσφέρει τη δυνατότητα της αναδρομής. Ακόμη, υπάρχουν δύο ειδικοί στόχοι, οι μετα-στόχοι AND και OR. Για να επιτύχει ο στόχος AND πρέπει να επιτύχουν οι δύο υπο-στόχοι του, ενώ για να επιτύχει ο OR αρκεί να επιτύχει ένας από τους δύο υπο-στόχους του. Οι στόχοι OR χρησιμεύουν σαν *σημεία επιλογής*, σημεία δηλαδή όπου διακλαδώνεται το δένδρο αναζήτησης.

Με τη βοήθεια των στόχων, λοιπόν, μπορούμε να υλοποιήσουμε τις μεθόδους αναζήτησης που επιθυμούμε. Η συνήθης μορφή του MAC είναι ενσωματωμένη στον Naxos. Στο εξής θα την καλούμε DFS, αφού εξερευνά το δένδρο αναζήτησης ακριβώς όπως η DFS, εφαρμόζοντας όμως ένα βήμα διάδοσης περιορισμών μετά από κάθε ανάθεση.

5 Η βιβλιοθήκη μεθόδων αναζήτησης Amorgos

Το κύριο μέρος της πτυχιακής εργασίας αυτής ήταν η ανάπτυξη μίας αντικειμενοστραφούς βιβλιοθήκης με μεθόδους αναζήτησης προσαρμοσμένες στα προβλήματα ικανοποίησης περιορισμών. Οι μέθοδοι αυτές προτάθηκαν, τα τελευταία χρόνια κυρίως, από διάφορους ερευνητές του χώρου της Τεχνητής Νοημοσύνης. Στην καρδιά της βιβλιοθήκης βρίσκεται ο επιλυτής Naxos. Για να συνεχιστεί η παράδοση της ονοματοδοσίας με βάση νησιά των Κυκλάδων η βιβλιοθήκη πήρε το όνομα της πανέμορφης Αμοργού.

Οι ήδη υπάρχουσες μέθοδοι αναζήτησης που έχουν υλοποιηθεί είναι οι εξής: ένα-δείγμα (onesamp) [6], επαναληπτική δειγματοληψία (iterative sampling) [6], αναζήτηση πρώτα κατά βάθος (Depth First Search), αναζήτηση με επαναληπτική διεύρυνση (iterative broadening search) [4], αναζήτηση με φραγμένη οπισθοδρόμηση (bounded backtrack search) [6], αναζήτηση φραγμένου βάθους με οπισθοδρόμηση (depth-bounded backtrack search) [1], αναζήτηση διαμοιρασμού πίστωσης (credit search) [1], αναζήτηση με περιορισμένες αναθέσεις (limited assignment number) [3], αναζήτηση πρώτα κατά βάθος με επανεκκινήσεις (DFS with restarts) [6], αναζήτηση με περιορισμένη ασυμφωνία (limited discrepancy search) [7], αναζήτηση με φραγμένη κατά βάθος ασυμφωνία (depth-bounded discrepancy search) [17].

5.1 Δύο νέες μέθοδοι αναζήτησης

Σε αυτήν την ενότητα παρουσιάζουμε δύο νέες ιδέες για μεθόδους αναζήτησης. Και οι δύο βασίζονται στον περιορισμό των τιμών που εξερευνώνται για κάθε μεταβλητή στην οποία ανατίθεται τιμή σε ένα συγκεκριμένο βάθος του δένδρου αναζήτησης.

Οι μέθοδοι θα επιβάλλουν ένα πλάτος w_v σε κάθε κόμβο v του δένδρου αναζήτησης, δηλαδή ένα άνω φράγμα στον αριθμό των παιδιών που επιτρέπεται να εξερευνηθούν. Συχνά οι μέθοδοι αναζήτησης θα καθορίζουν το πλάτος με βάση το ποσοστιαίο πλάτος. Το ποσοστιαίο πλάτος p_d σε βάθος d αφορά σε μεταβλητές που

αντιστοιχούν σε κόμβους βάθους d , και καθορίζει το ποσοστό των τιμών των μεταβλητών αυτών που θα εξερευνηθούν. Σε αυτήν την περίπτωση το πλάτος μπορεί να προκύψει ως εξής:

$$w_v = \lceil p_{d(v)} \cdot |D_{x(v)}| \rceil$$

για οποιονδήποτε κόμβο του δένδρου αναζήτησης. Στην παραπάνω εξίσωση $d(v)$ είναι το βάθος του κόμβου v στο δένδρο και $x(v)$ είναι η μεταβλητή που αντιστοιχεί στον κόμβο v .

Αναζήτηση με σταδιακό περιορισμό πλάτους Η Αναζήτηση με σταδιακό περιορισμό πλάτους (Gradual narrowing search - GNS) στηρίζεται στην ίδια ιδέα με την DBS, δηλαδή στο γεγονός ότι η αξιοπιστία του ευριστικού επιλογής τιμής αυξάνει με το βάθος. Αντί όμως να επιλέγεται ένα όριο βάθους πάνω από το οποίο εξερευνώνται όλες οι τιμές μίας μεταβλητής και κάτω από το οποίο εξερευνάται μόνο μία, η μετάβαση γίνεται πιο βαθμιαία: Όλες οι τιμές της ρίζας εξερευνώνται. Στη συνέχεια το ποσοστιαίο πλάτος μειώνεται γραμμικά ως προς το βάθος. Στα φύλλα το πλάτος γίνεται 1.

Αναζήτηση με συναρτησιακό περιορισμό πλάτους Είδαμε ότι η GNS μειώνει το ποσοστιαίο πλάτος γραμμικά με το βάθος. Γιατί όμως να είναι γραμμική αυτή η μείωση; Μπορεί σε κάποια προβλήματα η γραμμική μείωση να δουλεύει καλά, μπορεί σε κάποια άλλα όμως κάποια άλλη συνάρτηση να έχει ως αποτέλεσμα μία πιο αποδοτική μέθοδο αναζήτησης. Η αναζήτηση με συναρτησιακό περιορισμό πλάτους (functional narrowing search - FNS) δίνει τη δυνατότητα στο χρήστη να ορίσει ο ίδιος τη συνάρτηση που θα καθορίζει το πλάτος ή το ποσοστιαίο πλάτος για οποιοδήποτε βάθος.

Η συνάρτηση πλάτους f παίρνει τρία ακέραια ορίσματα. Το πρώτο είναι το μέγεθος του πεδίου της μεταβλητής s που αντιστοιχεί στον τρέχοντα κόμβο, το δεύτερο είναι το τρέχον βάθος της αναζήτησης d και το τρίτο είναι ο αριθμός των επιπέδων του δένδρου αναζήτησης, δηλαδή ο αριθμός n των μεταβλητών. Θα πρέπει $s \geq 1$ και $0 \leq d < n$. Επιστρέφεται ένας ακέραιος, το πλάτος για τον τρέχοντα κόμβο. Θα πρέπει να ισχύει $1 \leq f(s, d, n) \leq s$

Ας δούμε μερικά παραδείγματα συναρτήσεων πλάτους. Αν χρησιμοποιηθεί η συνάρτηση

$$f_{\text{onesamp}}(s, d, n) = 1$$

τότε η FNS γίνεται ισοδύναμη με την onesamp. Η συνάρτηση

$$f_{BBS}(s, d, n) = \begin{cases} s & \text{if } d < C; \\ 1 & \text{otherwise.} \end{cases}$$

μετατρέπει την FNS σε DBS με όριο βάθους C . Παρόμοια, αν χρησιμοποιήσουμε τη συνάρτηση

$$f_{IBROAD}(s, d, n) = C$$

παίρνουμε μία επανάληψη της iterative broadening, με $breadthLimit = C$. Πολύ εύκολα μπορούμε να προσομοιώσουμε και την GNS, αν χρησιμοποιήσουμε την παρακάτω συνάρτηση:

$$f_{GNS}(s, d, n) = \left\lceil \left(1 - \frac{d}{n}\right) s \right\rceil$$

Θεωρούμε ότι κάθε κλήση της συνάρτησης πλάτους f παίρνει σταθερό χρόνο. Επίσης έστω ότι $|D_x| \leq k$ για οποιαδήποτε μεταβλητή x και επίσης

$$\forall i, k_1, k_2, n \quad k_1 > k_2 \Rightarrow f(i, k_1, n) \geq f(i, k_2, n)$$

Τότε, η χρονική πολυπλοκότητα της FNS είναι η εξής:

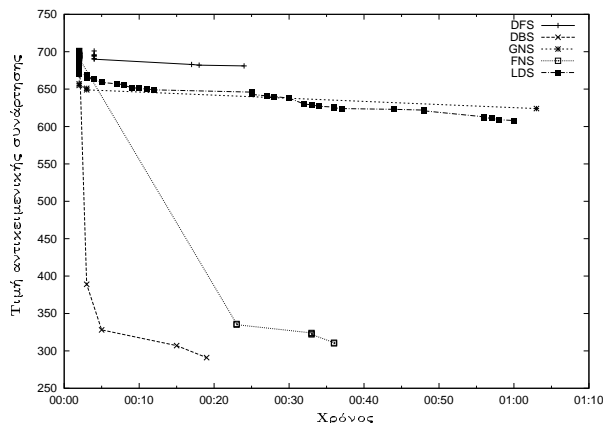
$$T(n, k) = O\left(\prod_{i=0}^{n-1} f(i, k, n)\right) \quad (1)$$

Παρατηρήστε ότι αν η $f(i, k, n)$ είναι μεγαλύτερη της μονάδας μόνο για ένα σταθερό αριθμό τιμών του i , τότε η πολυπλοκότητα της FNS είναι πολυωνυμική.

6 Μετρήσεις απόδοσης της Amorgos

Για τη σύγκριση των διαφόρων μεθόδων αναζήτησης που βρίσκονται υλοποιημένες στην Amorgos χρησιμοποιήθηκαν διάφορα προβλήματα. Κάποια από αυτά φάνηκε ότι δεν ήταν αρκετά «δύσκολα» ώστε να κάνουν ένα σαφή διαχωρισμό των μεθόδων. Το τοπίο όμως ξεκαθάρισε με την αξιοποίηση προβλημάτων που απαντώνται στον πραγματικό κόσμο.

Η κατάστρωση ωρολογίου προγράμματος είναι ένα πρόβλημα του πραγματικού κόσμου στο οποίο μπορούν να φανούν αρκετά οι διαφορές μεταξύ των μεθόδων αναζήτησης. Τα στιγμιότυπα που εξετάζονται προέρχονται από το διεθνή διαγωνισμό ITC (International Timetabling Competition) 2008, στον οποίον υπήρξε και συμμετοχή του επιλυτή Naxos σε συνδυασμό με τη βιβλιοθήκη Amorgos. Όπως φαίνεται στο Σχήμα 1, η DFS δεν αποδίδει καθόλου σε σχέση με τις υπόλοιπες μεθόδους. Ξεκάθαρος νικητής για αυτό το πρόβλημα είναι η DBS, ακολουθούμενη από την FNS. Η συνάρτηση πλάτους με την οποία εφοδιάσαμε την FNS εξερευνά όλα τα παιδιά του τρέχοντος κόμβου αν ο κόμβος αυτός δεν απέχει πολύ από τη ρίζα ή τα φύλλα. Αλλιώς εξερευνά μόνο ένα παιδί. Αξίζει να επισημανθεί εδώ ότι η συνάρτηση πλάτους αυτή ήταν ένα από τα τρία δείγματα συναρτήσεων που αναπτύχθηκαν ως παραδείγματα. Μοιραία θα υπάρχουν συναρτήσεις πλάτους που θα αποδίδουν ακόμη καλύτερα. Έτσι, βγάζουμε το συμπέρασμα ότι η μέθοδος FNS αξίζει να διερευνηθεί περαιτέρω.



Σχήμα 1. Σύγκριση μεθόδων αναζήτησης σε ένα στιγμιότυπο κατάρτισης ωρολόγιου προγράμματος

7 Συμπεράσματα και μελλοντική δουλειά

Η συνεισφορά της εργασίας αυτής μπορεί να συνοψισθεί στα εξής δύο σημεία: (i) Δημιουργήθηκε η βιβλιοθήκη αμοργός σαν επέκταση του επιλυτή Naxos και (ii) εισήχθησαν δύο νέες ελπιδοφόρες μέθοδοι αναζήτησης. Οι δύο νέες μέθοδοι αναζήτησης, η αναζήτηση με σταδιακό περιορισμό πλάτους (GNS) και η γενίκευσή της, η αναζήτηση με συναρτησιακό περιορισμό πλάτους (FNS), έδωσαν αρκετά ενθαρρυντικά αποτελέσματα. Πιθανώς μελλοντικές μετρήσεις σε άλλα προβλήματα θα δείξουν την υπεροχή της μιας ή της άλλης σε συγκεκριμένα πεδία εφαρμογής. Ειδικά για την FNS, είναι σίγουρο ότι οι δυνατότητές της δεν έχουν εξαντληθεί. Ένα ερώτημα είναι το ποια είναι η συνάρτηση αυτή που εξασφαλίζει τη βέλτιστη συμπεριφορά της FNS για κάποιο συγκεκριμένο πρόβλημα. Δε φαίνεται δύσκολο να μπορεί να δημιουργηθεί κάποια μέθοδος που να κατασκευάζει τη βέλτιστη αυτή συνάρτηση ή τουλάχιστον να την προσεγγίζει.

Τέλος, χρήσιμες επεκτάσεις της βιβλιοθήκης θα ήταν ο εμπλουτισμός της με μεθόδους τοπικής αναζήτησης και ο συνδυασμός των υλοποιημένων αλγορίθμων για τη δημιουργία υβριδικών μεθόδων αναζήτησης.

Ευχαριστίες Καταρχάς θα ήθελα να ευχαριστήσω τον καθηγητή μου Παναγιώτη Σταματόπουλο. Είχε την υπομονή και τη διάθεση να με καθοδηγήσει μέσω συζητήσεων και να προσφέρει άμεσες και κατατοπιστικότερες απαντήσεις σε οποιαδήποτε απορία κι αν είχα.

Επίσης ανεκτίμητη ήταν η συμβολή του συναδέλφου μου Νίκου Ποθητού, ο οποίος χωρίς καμία υποχρέωση και με μεγάλη προθυμία έκανε το παν για να με συμβουλευεί και να με βοηθάει σε κάθε μου πρόβλημα, ακόμη και κατά τη διάρκεια της στρατιωτικής του θητείας. Του οφείλω κατά μεγάλο ποσοστό την ποιότητα της εργασίας αυτής.

Αναφορές

1. Barták, R.: *Incomplete Depth-First Search Techniques: A Short Survey* In *Proceedings of 6th Workshop on Constraint Programming for Decision and Control (CPDC2004)*, Gliwice, Poland, 2004.
2. Barták, R.: *Online guide to constraint programming*. Διαθέσιμο στη διεύθυνση <http://kti.ms.mff.cuni.cz/~bartak/constraints/>.
3. Barták, R. and Rudová, H.: *Limited Assignments: A New Cutoff Strategy for Incomplete Depth-First Search* In *Proceedings of the 2005 ACM symposium on Applied computing*, 388–392, 2005.
4. Ginsberg, M and Harvey, W: *Iterative Broadening In Artificial Intelligence*, Volume 55, Issue 2–3, 367–383, 1992
5. Glover, F.: *Tabu Search — Part I* In *Orsa Journal on Computing*, 190–206, 1989 1: 3.
6. Harvey, W: *Nonsystematic Backtracking Search* Phd thesis, CIRL, University of Oregon, 1269 University of Oregon; Eugene, OR USA 97403-1269, 1995.
7. Harvey, W.D. and Ginsberg, M.L.: *Limited discrepancy search*, in *Proc. 14th Int. Joint Conf. on Artificial Intelligence*, (IJCAI-95), 607–615, 1995.
8. Korf, R: *Improved limited discrepancy search*, in *Proceedings of AAAI-96*, 288–291, 1996.
9. Lewis, R: *Metaheuristics can solve sudoku puzzles*, In *Journal of Heuristics* 13, 4 (Aug. 2007), 387-401.
10. Mackworth, A. K. and Freuder, E. C.: *The complexity of constraint satisfaction revisited*, In *Artificial intelligence in Perspective*, D. G. Bobrow, Ed. MIT Press, Cambridge, MA, 57–62, 1994.
11. Rossi, F., Petrie, C. and Dhar, V.: *On the Equivalence of Constraint Satisfaction Problems*, In *Proceedings of the 9th European Conference on Artificial Intelligence*, (ECAI-90), 550–556, 1990.
12. Rudová, H. and Veřmiřovský, K.: *Random Placement Problem (RPP)*, Διαθέσιμο στη διεύθυνση <http://www.fi.muni.cz/~hanka/rpp/spec.html>.
13. Russell, S. and Norvig, P.: *Artificial Intelligence A Modern Approach*, Prentice Hall, 2003.
14. Sabin, D. and Freuder, E.C.: *Contradicting conventional wisdom in constraint satisfaction*, In *Proceedings of ECAI-94*, 125–129, 1994.
15. Sellmann, M: *The Social Golfer Problem* Διαθέσιμο στη διεύθυνση <http://www.cs.brown.edu/~sello/golf.html>.
16. Sasic, R. and Gu, J.: *A polynomial time algorithm for the n-queens problem*, In *SIGART*, 1(3), 1990.
17. Walsh, T: *Depth-bounded discrepancy search*, In *Proceedings of 15th IJCAI*, 1997.
18. Yato, T. and Seta, T: *Complexity and Completeness of Finding Another Solution and Its Application to Puzzles* In *IEICE Trans. Fundamentals*, Vol. E86-A, No. 5, pp. 1052–1060, 2003.
19. Ποθητός, Ν.: *Naxos Solver: Εγχειρίδιο Χρήσης* Διαθέσιμο στη διεύθυνση <http://www.di.uoa.gr/~grad0780/naxos/naxos.pdf>.
20. Ποθητός, Ν.: *Αυτόματη κατασκευή ωρολογίων προγραμμάτων μέσω προγραμματισμού με περιορισμούς* Πτυχιακή εργασία, Τμήμα Πληροφορικής και Τηλεπικοινωνιών, ΕΚΠΑ, 2005.
21. Σταματόπουλος, Π.: *Σημειώσεις λογικού προγραμματισμού*, Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών, 2005.

Πλαίσιο Επεξεργασίας και Ανάλυσης Υπερηχογραφικής Εικόνας

Ευστράτιος Γ. Κεραμιδάς *

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών
Τμήμα Πληροφορικής και Τηλεπικοινωνιών
Πανεπιστημιούπολη, 157 84 Αθήνα

Περίληψη Σε αυτή την εργασία παρουσιάζεται ένα ολοκληρωμένο πλαίσιο επεξεργασίας και ανάλυσης εικόνας. Το πλαίσιο αυτό αναπτύχθηκε με στόχο την εξερεύνηση συγχρόνων μεθοδολογιών επεξεργασίας και ανάλυσης εικόνας σε πραγματικά υπερηχογραφικά δεδομένα. Η πειραματική εφαρμογή του σε πραγματικές υπερηχογραφικές εικόνες οδηγεί σε χρήσιμα συμπεράσματα για την απόδοση των αντίστοιχων μεθόδων.

Λέξεις Κλειδιά: Πλαίσιο Επεξεργασίας Εικόνας, Ανάλυση Εικόνας, Υπέρηχος, Θυρεοειδής, Μέθοδοι Εξαγωγής Χαρακτηριστικών.

1 Εισαγωγή

Στη σύγχρονη εποχή μοντέρνες τεχνικές απεικόνισης έχουν δημιουργηθεί, με κομβικό σημείο τα τέλη του 1800, όπου έγινε η ανακάλυψη των ακτινών X από τον Roentgen [1]. Από τότε μέχρι σήμερα πολλές πρακτικές λήψης, απεικόνισης και επεξεργασίας ιατρικών εικόνων έχουν προταθεί και εφαρμοστεί σε μεγάλη κλίμακα στην ιατρική πρακτική.

Στον τομέα της επεξεργασίας και ανάλυσης ιατρικής εικόνας, μια ποικιλία από τεχνολογίες επιτρέπει τη συνεχώς αυξανόμενη ευκρίνεια και ευαισθησία στην απεικόνιση οργάνων και ιστών. Πολλές είναι οι απεικονιστικές τεχνικές που έχουν εξελιχθεί τα τελευταία χρόνια συμπεριλαμβανομένης και της Υπερηχογραφίας (Ultrasonography - US). Η υπερηχογραφία είναι από τις πιο συνηθισμένες μεθόδους ιατρικής απεικόνισης στην καθημερινή ιατρική πρακτική. Περιλαμβάνει την έκθεση τμημάτων του ανθρώπινου σώματος σε ηχητικά κύματα υψηλής συχνότητας για τη δημιουργία εικόνων από την ανάκλασή τους. Αξίζει να σημειωθεί ότι αυτή η τεχνολογία δεν χρησιμοποιεί κανένα είδος επικίνδυνης ακτινοβολίας.

Εξελίξεις στην υπερηχογραφική τεχνολογία έχουν οδηγήσει σε καλύτερη ποιότητα εικόνας, σε σχέση με τις προγενέστερες γενιές υπερηχογράφων. Το κόστος και ο χρόνος που απαιτείται για να ληφθεί μια υπερηχογραφική εικόνα αν συγκριθούν με άλλες τεχνικές απεικόνισης, όπως ο μαγνητικός τομογράφος, κάνουν τον υπέρηχο μια ιδιαίτερα ελκυστική επιλογή. Σήμερα, η διαγνωστική εκτίμηση των

* Επιβλέπων: Δημήτρης Μαρούλης

ακτινολόγων με όργανα υπέρηχου βασίζεται κυρίως σε υπερηχογραφικές εικόνες δύο διαστάσεων (B-mode Ultrasonography).

Η ερμηνεία υπερηχογραφικών εικόνων αποτελούσε πάντοτε ιδιαίτερα περίπλοκο πρόβλημα, ακόμα και για τους πιο άρτια εκπαιδευμένους ιατρούς. Επειδή η αξιολόγηση των εικόνων πραγματοποιείται οπτικά, η ερμηνεία τους καθίσταται αρκετά υποκειμενική και επηρεάζεται από πολλούς παράγοντες. Μεταξύ αυτών είναι η προγενέστερη γνώση του αντικειμένου που εξετάζεται, η εκπαίδευση του γιατρού καθώς και η ποιότητα της συσκευής υπερηχογραφικής απεικόνισης.

Οι σύγχρονες μέθοδοι επεξεργασίας και ανάλυσης της ψηφιακής ιατρικής εικόνας στοχεύουν κυρίως στην εξαγωγή ποσοτικής πληροφορίας. Αυτή μπορεί να είναι με τη μορφή δυαδικής πληροφορίας (απουσία / παρουσία αντικείμενου / ιδιότητας) ή τη μέτρηση του μεγέθους ενός αντικείμενου, της θέσης του, του σχήματος του, ή στατιστικών πληροφοριών. Σε κάποιες εφαρμογές μπορεί να είναι δυνατή ή επιθυμητή, η αυτοματοποίηση ορισμένων μόνο σταδίων της ανάλυσης της ιατρικής εικόνας, αφήνοντας τα υπόλοιπα για ερμηνεία από τον ιατρό.

Στόχος αυτής της εργασίας αποτελεί η αξιολόγηση σύγχρονων τεχνικών επεξεργασίας εικόνας για τον χαρακτηρισμό τύπων ιστού του θυροειδούς αδένου. Προηγούμενες έρευνες έχουν δείξει ότι υπάρχουν τεχνικές ανάλυσης εικόνας που μπορούν να συνεισφέρουν στην αντικειμενικότερη διάγνωση διαφόρων νόσων του θυροειδούς αδένου με βάση υπερηχογραφικές εικόνες. Για την αξιολόγηση των τεχνικών αυτών δημιουργήσαμε ένα Πλαίσιο Επεξεργασίας & Ανάλυσης Υπερηχογραφικής Εικόνας (ΠΕΑΥΕ) που παρέχει ένα ολοκληρωμένο περιβάλλον εκτέλεσης πειραμάτων επεξεργασίας και ανάλυσης εικόνας. Με τη χρήση του ΠΕΑΥΕ, πραγματοποιήθηκε ένα μεγάλο σύνολο πειραμάτων και παρουσιάζονται αποτελέσματα που οδηγούν στην αξιολόγηση μεθόδων προ-επεξεργασίας εικόνας, εξαγωγής χαρακτηριστικών και ταξινόμησης χαρακτηριστικών σε υπερηχογραφικές εικόνες.

Η συνέχεια της παρούσας εργασίας δομείται σε τέσσερις παραγράφους. Στη δεύτερη παράγραφο γίνεται μια παρουσίαση παλαιότερων αλλά και σύγχρονων επιστημονικών εργασιών με στόχο την εξαγωγή χαρακτηριστικών, χρήσιμων στην ιατρική υπερηχογραφική διάγνωση. Στην τρίτη παράγραφο παρουσιάζεται η δομή, τα χαρακτηριστικά και ορισμένες βασικές πληροφορίες για το ΠΕΑΥΕ. Στη τελευταία παράγραφο παρουσιάζεται και σχολιάζεται το σύνολο των αποτελεσμάτων των πειραμάτων που εκτελέστηκαν σε υπερηχογραφικές εικόνες του θυροειδή αδένου.

2 Μέθοδοι Εξαγωγής Χαρακτηριστικών από Υπερηχογραφικές Εικόνες

Από τις αρχές της δεκαετίας του 1980 μέχρι σήμερα, παρουσιάστηκαν πολλές μελέτες με σκοπό την ανάπτυξη και τη βελτιστοποίηση μεθόδων για την εξαγωγή χαρακτηριστικών, ικανών να κωδικοποιήσουν την πληροφορία που περιέχεται σε υπερηχογραφικές εικόνες. Σε αυτή την παράγραφο παρουσιάζονται οι πιο σημαντικές από αυτές.

2.1 Χαρακτηριστικά Ιστογράμματος

Το ιστόγραμμα αποτελεί τη βάση για πολλές τεχνικές επεξεργασίας εικόνας. Όσον αφορά τις μεθόδους εξαγωγής χαρακτηριστικών, σε πολλές ερευνητικές εργασίες όπως στην [2] έχουν προταθεί και εφαρμοστεί μέτρα, που προκύπτουν από το ιστόγραμμα όπως η μέση τιμή, η τυπική απόκλιση, η κύρτωση, η ενέργεια και η εντροπία.

2.2 Χαρακτηριστικά Μετασχηματισμού Κυματιδίου

Παρά το γεγονός ότι ο μετασχηματισμός Fourier (Fourier Transform-FT) χρησιμοποιείται για την επεξεργασία εικόνας από το 1950, ένας πιο σύγχρονος μετασχηματισμός που κερδίζει συνεχώς έδαφος, είναι ο μετασχηματισμός κυματιδίου (Wavelet Transform-WT). Αντίθετα με το μετασχηματισμό Fourier, του οποίου οι βασικές συναρτήσεις είναι ημιτονοειδείς, ο μετασχηματισμός κυματιδίου βασίζεται σε μικρά κύματα, που καλούνται κυματίδια (wavelets), με μεταβλητή συχνότητα και περιορισμένη διάρκεια. Ιδιαιτερότητα του WT είναι ότι εμπεριέχει σε μεγάλο βαθμό το χαρακτηριστικό της τοπικότητας. Αυτό σημαίνει πως δεν μετατρέπει μία καθαρά χρονική ή χωρική περιγραφή σε μία καθαρά συχνοτική περιγραφή, αλλά οδηγεί σε μια χώρο-συχνοτική ή χρόνο-συχνοτική περιγραφή.

Ποικιλία χαρακτηριστικών για υπερηχογραφικές εικόνες που βασίζονται στα αποτελέσματα του μετασχηματισμού κυματιδίου, έχουν προταθεί στη βιβλιογραφία όπως η [3]. Ορισμένα από τα πιο αποτελεσματικά χαρακτηριστικά για υπερηχογραφικές εικόνες είναι η μέση τιμή, η διασπορά, η απόκλιση αντίθεσης (variance contrast), η αυτοσυσχέτιση αντίθεσης (autocorrelation contrast), η κατανομή παραμόρφωσης (distribution distortions), η κύρτωση (kurtosis) και ο λόγος ενέργειας (energy ratio).

2.3 Χαρακτηριστικά Πίνακα Συνεμφάνισης

Ο πίνακας συνεμφάνισης (Co-occurrence Matrix-CM), χρησιμοποιείται για να περιγράψει πρότυπα από γειτονικά εικονοστοιχεία σε μια εικόνα, με συγκεκριμένες αποστάσεις μεταξύ τους. Για τον υπολογισμό χαρακτηριστικών υψής από πίνακες συνεμφάνισης, απαιτείται η δημιουργία 4 τέτοιων πινάκων, έναν για κάθε κατεύθυνση. Από κάθε πίνακα συνεμφάνισης μπορούν να υπολογιστούν στατιστικά μέτρα, τα οποία να οδηγήσουν στο χαρακτηρισμό της υψής μιας εικόνας. Τα πιο σημαντικά χαρακτηριστικά μέτρα είναι αυτά που πρότεινε το 1979 ο Haralick [4].

2.4 Χαρακτηριστικά Muzzolini

Στην εργασία των Russell Muzzolini, Yee-Hong Yang και Roger Pierson [5] το 1993, προτάθηκαν 32 χαρακτηριστικά, που παρουσιάστηκαν και αξιολογήθηκαν για χαρακτηρισμό υψής υπερηχογραφικών εικόνων. Η μέθοδος που προτάθηκε αποτελείται από 21 χαρακτηριστικά στο πεδίο του χώρου και 11 στο πεδίο των συχνοτήτων.

2.5 Χαρακτηριστικά Τοπικού Δυαδικού Πρότυπου

Η μέθοδος του Τοπικού Δυαδικού Προτύπου (Local Binary Pattern-LBP) [6], προήλθε από το μοντέλο των Wang και He [7] που ονομάζεται μονάδα υφής (Texture Unit). Στο βασικό μοντέλο LBP κάθε μονάδα υφής αποτελείται από εννιά εικονοστοιχεία, που ανήκουν σε μια γειτονία 3×3 εικονοστοιχείων. Η μέθοδος εφαρμόζεται σε όλες τις δυνατές περιοχές της εικόνας, παράγοντας ισάριθμους κωδικούς LBP, οι οποίοι στη συνέχεια τοποθετούνται σε κατανομές οι οποίες και σχηματίζουν τα διανύσματα χαρακτηριστικών. Αρκετές μελέτες της βιβλιογραφίας έχουν βασιστεί στο μοντέλο LBP, για τον χαρακτηρισμό ιστού σε υπερηχογραφικές εικόνες όπως [8].

3 Πλαίσιο Επεξεργασίας και Ανάλυσης Υπερηχογραφικής Εικόνας

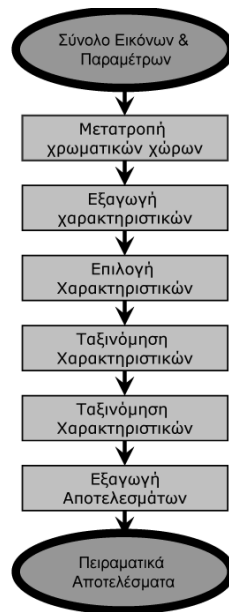
Σκοπός αυτής της μελέτης, είναι η σύγκριση μεθόδων επεξεργασίας εικόνας, εξαγωγής και ταξινόμησης χαρακτηριστικών σε υπερηχογραφικές εικόνες. Για τη σύγκριση απαιτείται η εκτέλεση μεγάλου πλήθους σύνθετων πειραμάτων, αποτελούμενων από διακριτές φάσεις, με πλήθος διαφορετικών παραμέτρων. Η ανάπτυξη του Πλαισίου Επεξεργασίας & Ανάλυσης Υπερηχογραφικής Εικόνας (ΠΕΑΥΕ) έγινε με σκοπό την ταχύτερη πραγματοποίηση και την καλύτερη οργάνωση μιας τέτοιας μελέτης.

3.1 Δομή

Τα πειράματα επεξεργασίας και ανάλυσης εικόνας αποτελούνται από τις παρακάτω γενικές φάσεις: προεπεξεργασία εικόνας, εξαγωγή, επιλογή και ταξινόμηση χαρακτηριστικών. Σε κάθε φάση, ανάλογα με τη μέθοδο που εφαρμόζεται, υπάρχουν παράμετροι που επηρεάζουν τη συμπεριφορά και τα αποτελέσματα των πειραμάτων. Οι παράμετροι που παίζουν ενεργό ρόλο σε κάθε πείραμα είναι αρκετοί, ενώ σκοπό των πειραμάτων αποτελεί, εκτός των άλλων, η εξερεύνηση των βέλτιστων τιμών για όλες τις εμπλεκόμενες παραμέτρους. Σε αυτό το πρόβλημα, το ΠΕΑΥΕ προσφέρει τη δυνατότητα ταχύτατης οργάνωσης και εκτέλεσης πολυπληθών συνόλων πειραμάτων, για όλους τους δυνατούς συνδυασμούς, όλων των μεταβλητών που επιθυμεί ο χρήστης, με τη δημιουργία ενός και μόνο αρχείου παραμέτρων.

Το ΠΕΑΥΕ αποτελείται από έξι διακριτά και ανεξάρτητα τμήματα (Σχ. 1). Καθένα από αυτά τα τμήματα ενσωματώνει μία λειτουργία, η οποία αποτελεί ένα σημαντικό κομμάτι των πειραμάτων επεξεργασίας και ανάλυσης εικόνας. Τα τμήματα αυτά είναι τα εξής:

- **Μετατροπή εικόνων σε κατάλληλο χρωματικό χώρο.** Ενσωματώνονται μέθοδοι μετατροπής εικόνων από ένα χρωματικό χώρο σε κάποιον άλλον συμπεριλαμβανομένων των RGB, XYZ, HSV και YIQ.
- **Κανονικοποίηση Εικόνων.** Ενσωματώνονται μέθοδοι κανονικοποίησης, συμπεριλαμβανομένων της μεθόδου κανονικοποίησης έντασης εικόνας και της μεθόδου ενίσχυσης αντίθεσης.



Σχήμα 1. Βασική Δομή Πλαίσιου Επεξεργασίας και Ανάλυσης Υπερχογραφικής Εικόνας.

- **Εξαγωγή Χαρακτηριστικών.** Ενσωματώνονται μέθοδοι εξαγωγής χαρακτηριστικών και παρέχεται επιπλέον η δυνατότητα εύκολης ενσωμάτωσης νέων μεθόδων. Μεταξύ των μεθόδων που έχουν ενσωματωθεί είναι και η μέθοδος Τοπικού Δυαδικού Προτύπου (LBP), η μέθοδος χαρακτηριστικών Krishnan, η μέθοδος χαρακτηριστικών μετασχηματισμού κυματιδίου, και η μέθοδος χαρακτηριστικών του πίνακα συνεμφάνισης.
- **Επιλογή Χαρακτηριστικών.** Ενσωματώνονται μέθοδοι για την επιλογή χαρακτηριστικών, όπως γενετικός αλγόριθμος επιλογής και μέθοδος Σειριακής προς τα Εμπρός Επιλογής με χρήση ταξινομητή k-Κοντινότερων Γειτόνων SFFS- k-NN.
- **Ταξινόμηση Χαρακτηριστικών.** Το σύνολο των χαρακτηριστικών που έχουν εξαχθεί και επιλεγεί, χρησιμοποιείται για να ελεγχθεί η δυνατότητα τους να διαχωρίζουν διαφορετικές κλάσεις. Σε αυτό το τμήμα λοιπόν ενσωματώνονται μέθοδοι ταξινόμησης χαρακτηριστικών, όπως Μηχανές Ανυσμάτων Στηρίξης(SVM) και μέθοδος κ-Κοντινότερων Γειτόνων.
- **Στατιστική Επεξεργασία Αποτελεσμάτων.** Εφαρμόζονται μέθοδοι με σκοπό την καλύτερη δυνατή αναπαράσταση των πειραματικών αποτελεσμάτων, όπως η μέθοδος για τη δημιουργία Πίνακα Σύγχυσης (Confusion Matrix), ενώ υπάρχει και δυνατότητα οπτικής αναπαράστασης των αποτελεσμάτων με τη δημιουργία τεχνητών εικόνων που αναπαριστούν χρωματικά τα αποτελέσματα ταξινόμησης.

Το ΠΕΑΥΕ αναπτύχθηκε σε γλώσσα προγραμματισμού ANSI C, για να είναι δυνατή και εύκολη η χρήση του σε διαφορετικά υπολογιστικά συστήματα (hardware) και με διαφορετικά λειτουργικά συστήματα (software).

3.2 Πρόσθετα Χαρακτηριστικά

Εκτός από τις βασικές λειτουργίες που περιγράφηκαν παραπάνω, στο ΠΕΑΥΕ έχουν ενσωματωθεί ορισμένα πρόσθετα χαρακτηριστικά που διευκολύνουν και επιταχύνουν τη διαδικασία εκτέλεσης πειραμάτων. Αυτά τα χαρακτηριστικά είναι τα εξής:

- **Αποστολή αποτελεσμάτων με μηνύματα ηλεκτρονικού ταχυδρομείου:** Παρέχεται η δυνατότητα αυτόματης αποστολής μηνυμάτων ηλεκτρονικού ταχυδρομείου με τα αποτελέσματα των πειραμάτων.
- **Επαναφορά εκτέλεσης συνόλου πειραμάτων:** Το ΠΕΑΥΕ είναι σχεδιασμένο να διαχειρίζεται πολυπληθή σύνολα πειραμάτων. Στην περίπτωση που η εκτέλεση ενός συνόλου διακοπεί, υπάρχει η δυνατότητα για αυτόματη επανεκκίνηση των πειραμάτων από το σημείο που έγινε η διακοπή εξοικονομώντας χρόνο επεξεργασίας.
- **Επαναχρησιμοποίηση χαρακτηριστικών:** Για την εξοικονόμηση υπολογιστικού χρόνου, παρέχεται η δυνατότητα επαναχρησιμοποίησης των χαρακτηριστικών που έχουν παραχθεί από προηγούμενες εκτελέσεις.
- **Επαναχρησιμοποίηση χρωματικά μετασχηματισμένων εικόνων:** Παρέχεται η δυνατότητα επαναχρησιμοποίησης χρωματικά μετασχηματισμένων εικόνων, που έχουν παραχθεί σε προηγούμενες εκτελέσεις, για την εξοικονόμηση υπολογιστικού χρόνου.
- **Συμπίεση αρχείων χαρακτηριστικών:** Το ΠΕΑΥΕ παρέχει τη δυνατότητα συμπίεσης των αρχείων που παράγει, έτσι ώστε να αποθηκεύονται αποδοτικότερα.
- **Διαγραφή παλαιών αρχείων:** Παρέχεται η δυνατότητα αυτόματης οριστικής διαγραφής όλων των αρχείων που δημιουργούνται κατά τη διάρκεια μιας πειραματικής διαδικασίας, για την εξοικονόμηση πολύτιμου χώρου στα μέσα αποθήκευσης του συστήματος.

3.3 Σχέδια Επέκτασης

Πέρα από τις μεθόδους που έχουν ενσωματωθεί στο ΠΕΑΥΕ, υπάρχει η δυνατότητα αλλά και η διαρκής ανάγκη για εισαγωγή νέων, σύγχρονων μεθόδων σε κάθε τμήμα του. Η εισαγωγή νέων μεθόδων μπορεί να γίνει εύκολα, αφού η αρχιτεκτονική του ΠΕΑΥΕ είναι δομημένη προς αυτή την κατεύθυνση.

Μία ακόμα επέκταση του ΠΕΑΥΕ, θα μπορούσε να στοχεύει στη χρήση του για ανάλυση και επεξεργασία, όχι μόνο εικόνων αλλά και βίντεο σε πραγματικό χρόνο.

4 Πειραματική Αξιολόγηση Μεθόδων

Σε αυτή την παράγραφο περιγράφεται η πειραματική διαδικασία εφαρμογής του ΠΕ-ΑΥΕ σε πραγματικές υπερηχογραφικές εικόνες του θυρεοειδούς αδένου και αξιολογούνται διαφορετικές μέθοδοι επεξεργασίας και ανάλυσης εικόνας από τα αντίστοιχα πειραματικά αποτελέσματα.

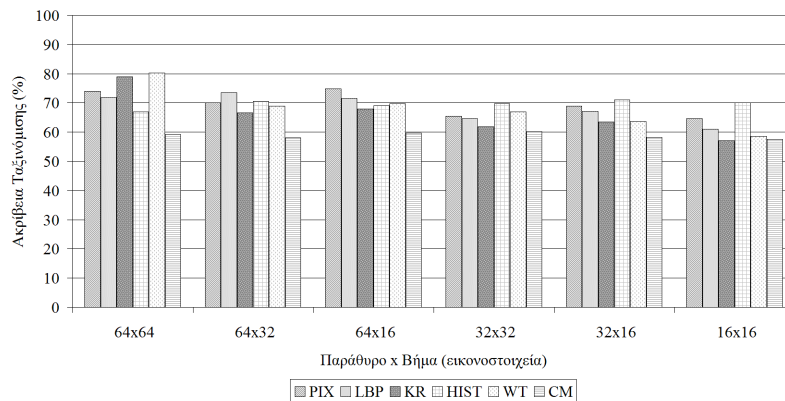
Πραγματοποιήθηκαν πειράματα σε ένα σύνολο από πραγματικές υπερηχογραφικές εικόνες του θυρεοειδούς αδένου. Αυτό αποτελείται από 145 εικόνες διαστάσεων 256×256 εικονοστοιχείων, οι οποίες προέρχονται από αναλογικά βίντεο υπερηχογραφικών εξετάσεων του θυρεοειδούς αδένου που ψηφιοποιήθηκαν. Οι εικόνες αυτές απεικονίζουν θυρεοειδείς αδένες που εμφανίζουν κανέναν, έναν ή και περισσότερους όζους.

Κάθε υπερηχογραφική εικόνα αξιολογήθηκε και διαγραμματίστηκε από έμπειρους ακτινολόγους, με σκοπό τη διάκριση μεταξύ υγιούς ιστού και όζου. Για τη διεξαγωγή των πειραμάτων ήταν απαραίτητος ο διαχωρισμός των εικόνων σε δύο σύνολα. Το πρώτο για την εκπαίδευση του υπό εξέταση συστήματός και το δεύτερο για τη δοκιμή του. Οι εικόνες χωρίστηκαν θεωρώντας ότι οι μισές ανήκουν στο σύνολο εκπαίδευσης και οι υπόλοιπες στο σύνολο δοκιμής.

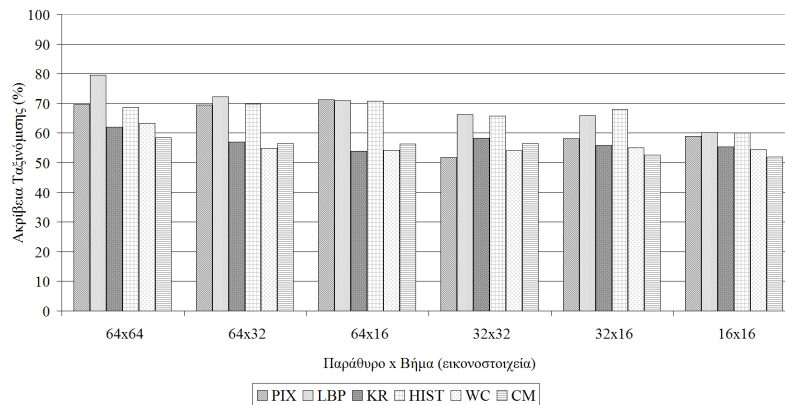
Τα πειράματα που εκτελέστηκαν περιλαμβάνουν την εξαγωγή χαρακτηριστικών από τετράγωνα παράθυρα που σαρώνουν ολόκληρη την κάθε εικόνα με διαφορετικές τιμές για το εύρος των παραθύρων (16 έως 64 εικονοστοιχεία) και για το βήμα μετατόπισης τους (16 έως 64 εικονοστοιχεία). Επιπλέον στα μισά πειράματα έχει εφαρμοστεί η μέθοδος κανονικοποίησης με ενίσχυση αντίθεσης [9], με ποσοστά εξαίρεσης 5, 10 και 20. Οι μέθοδοι εξαγωγής χαρακτηριστικών που χρησιμοποιήθηκαν είναι αυτές που βασίστηκαν στις απλές τιμές εικονοστοιχείων (Pixels-PIX), του Τοπικού Δυαδικού Πρότυπου (Local Binary Pattern-LBP), των χαρακτηριστικών Μετασχηματισμού Κυματιδίου (Wavelet Transform-WT), των χαρακτηριστικών (Krishnan-KR) [10] και των χαρακτηριστικών του Πίνακα Συνεμφάνισης (Co-occurrence Matrix-CM). Για την ταξινόμηση των χαρακτηριστικών χρησιμοποιήθηκαν οι αλγόριθμοι των Μηχανών Ανυσμάτων Στήριξης (SVM)[11] με γραμμικό και πολυωνυμικό πυρήνα, και του ταξινομητή k -Κοντινότερων Γείτονων (k -Nearest Neighbours k -NN)[12] με δύο τύπους αποστάσεων, ευκλείδεια και τομής ιστογραμμάτων.

Στα πειράματα δοκιμάστηκαν εξαντλητικοί συνδυασμοί διαφόρων τιμών των παραμέτρων των μεθοδολογιών που αναφέρονται παραπάνω. Τα βέλτιστα αποτελέσματα για κάθε τύπο ταξινομητή, για κάθε χαρακτηριστικό και για διαφορετικές τιμές βήματος και μεγέθους παραθύρων σάρωσης απεικονίζονται στα σχήματα 2, 3 και 4.

Για την περίπτωση του ταξινομητή SVM, όπως φαίνεται και από το σχήμα 2, την καλύτερη επίδοση, 80,1%, παρουσίασαν τα χαρακτηριστικά του μετασχηματισμού κυματιδίου (WT), για παράθυρο σάρωσης 64×64 εικονοστοιχείων και βήμα 64 εικονοστοιχείων. Για τον ταξινομητή k -NN (Σχ. 3 και 4), προκύπτει ότι η μέγιστη μέση ακρίβεια ταξινόμησης για αυτά τα χαρακτηριστικά είναι αρκετά χαμηλότερη, και πιο συγκεκριμένα δεν ξεπερνάει το 65% για με χρήση ευκλείδειας απόστασης και το 67% με χρήση απόστασης τομής ιστογραμμάτων. Από το σχήμα των συνολικών αποτελεσμάτων του ταξινομητή SVM, ως γενικότερο συμπέρασμα μπορεί



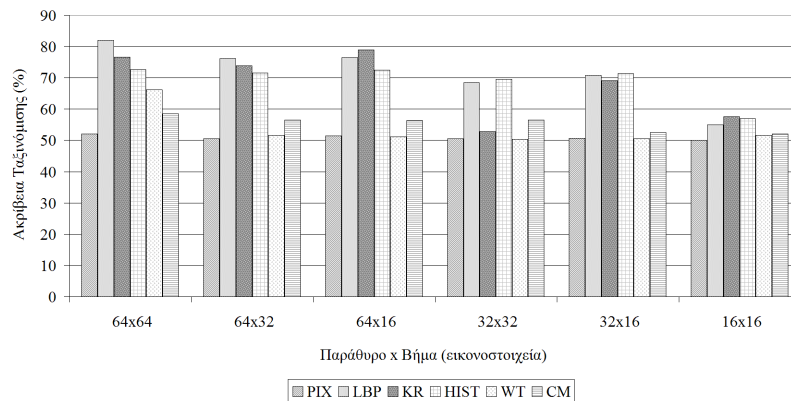
Σχήμα 2. Συνολικά βέλτιστα αποτελέσματα με τον ταξινομητή SVM.



Σχήμα 3. Συνολικά βέλτιστα αποτελέσματα με ταξινομητή k-NN με ευκλείδεια απόσταση.

να εξαχθεί, πως όλα τα χαρακτηριστικά που δοκιμάστηκαν, παράγουν καλύτερα αποτελέσματα μέγιστης μέσης ακρίβειας ταξινόμησης, για μεγάλα παραθύρα σάρωσης, άρα για πολλά εικονοστοιχεία σε κάθε παράθυρο. Επιπλέον από το σχήμα 2, προκύπτει ότι η μέγιστη μέση ακρίβεια ταξινόμησης μειώνεται καθώς το μέγεθος του παραθύρου σάρωσης μειώνεται.

Στο σχήμα 3 παρουσιάζονται τα συνολικά βέλτιστα αποτελέσματα της μέσης ακρίβειας ταξινόμησης, για όλες τις διαστάσεις παραθύρου και βήματος σάρωσης, για τον ταξινομητή k-NN με ευκλείδεια απόσταση. Τα αποτελέσματα υποδεικνύουν ως βέλτιστο διάλυμα χαρακτηριστικών, αυτό του Τοπικού Δυαδικού Πρότυπου (LBP). Τα χαρακτηριστικά LBP πετυχαίνουν βέλτιστη μέση ακρίβεια ταξινόμησης



Σχήμα 4. Συνολικά βέλτιστα αποτελέσματα με ταξινομητή k-NN με απόσταση τομής ιστογραμμάτων.

79.8%. Η αμέσως μικρότερη επίδοση για την ίδια διάσταση παραθύρου σάρωσης και βήματος σάρωσης, 70,6%, είναι σχεδόν δέκα ποσοστιαίες μονάδες χαμηλότερη και επιτυγχάνεται με τη μέθοδο των τιμών των εικονοστοιχείων (PIX).

Στα πειράματα που διεξήχθησαν με τον ταξινομητή k-NN με απόσταση τομής ιστογραμμάτων (Σχ. 4), την καλύτερη επίδοση σημείωσε και πάλι η μέθοδος εξαγωγής χαρακτηριστικών LBP με ακρίβεια 82%, με τετράγωνο παράθυρο σάρωσης διαστάσεων 64×64 εικονοστοιχείων και βήμα σάρωσης 64 εικονοστοιχείων. Επίσης, η μέθοδος που είναι βασισμένη στα χαρακτηριστικά Krishnan, πέτυχε αξιόλογα αποτελέσματα, αφού η ακρίβεια της έφτασε το 79%, με παράθυρο σάρωσης διαστάσεων 64×64 εικονοστοιχείων και βήμα σάρωσης 16 εικονοστοιχείων.

Τα συμπεράσματα που προέκυψαν από ολόκληρη την πειραματική διαδικασία σχετίζονται με την απόδοση των μεθόδων εξαγωγής χαρακτηριστικών, με τις μεθόδους ταξινόμησης αυτών αλλά και με την αποτελεσματικότητα χρήσης συγκεκριμένων ταξινομητών για συγκεκριμένα χαρακτηριστικά. Από τα παραπάνω πειράματα, το καλύτερο αποτέλεσμα ταξινόμησης που προέκυψε ήταν 82.6%, από τα χαρακτηριστικά της μεθόδου LBP, με ταξινομητή k-NN με απόσταση τομής ιστογραμμάτων και με εφαρμογή της μεθόδου κανονικοποίησης των εικόνων.

Τελικά αξίζει να σημειωθεί, ότι οι μέθοδοι που δημιουργούν ιστογράμματα σαν διανύσματα χαρακτηριστικών, έδειξαν να συνεργάζονται καλύτερα με τον ταξινομητή k-NN με απόσταση τομής ιστογραμμάτων. Για τις μεθόδους που δεν δημιουργούν ιστογράμματα, καλύτερες επιδόσεις παρουσίασε ο ταξινομητής SVM, ενώ ο k-NN με ευκλείδεια απόσταση αν και ταχύτερος και με λιγότερες παραμέτρους από τον SVM, στις περισσότερες περιπτώσεις υπολείπεται σε απόδοση

Αναφορές

- [1] Robb, R.A., (2005) “Biomedical Imaging: Past, Present and Predictions”, Proceedings of the First International Symposium on Intelligent Assistance in Diagnosis of Multi-Dimensional Medical Image, Japan.
- [2] Mougiakakou, S., Golemati, S., Gousias, I., Nicolaidis, A., and Nikita K., (2007) “Computer-Aided Diagnosis Of Carotid Atherosclerosis Based On Ultrasound Image Statistics, Laws’A Texture And Neural Networks”, *Ultrasound in Medicine & Biology*, Vol. 33, No. 1, pp. 26-36.
- [3] Ahmadian, A., Mostafa, A., Abolhassani, M.D., Riahi Alam, N., and Gitti, M., (2004) “An efficient texture feature extraction method for classification of liver sonography based on Gabor Wavelet”, *Medicon*.
- [4] Haralick, R., Shanmugam, K. and Dinstein, I., (1979) “Textural Features for Image Classification”, *IEEE Transactions on systems, Man, and Cybernetics*, pp. 610-621.
- [5] Muzzolini, R., Yang, Y., and Pierson, R., (1994) “Texture Characterization Using Robust Statistics”, *Pattern Recognition*, Vol. 27, No. 1, pp. 119-134.
- [6] Ojala, T., Pietikainen, M., and Harwood, D., (1996) “A Comparative Study Of Texture Measures With Classification Based On Feature Distributions”, *Pattern Recognition*, Vol. 29, No. 1, pp. 51-59.
- [7] Wang L., and He, D.C., (1990) “Texture classification using texture spectrum”, *Pattern Recognition* Vol. 23, pp. 905-910.
- [8] Rotger, D., Radeva, P., Rodriguez, O., Mauri, J., (2003) “Near Real-Time Plaque Segmentation of IVUS”, *Computers in Cardiology*, Vol. 30, pp. 69-72.
- [9] Lillesand, T., and Keefer, R., (2000) “Remote Sensing and Image Interpretation”, New York: John Wiley and Sons, pp. 724.
- [10] Tjoa, M. and Krishnan, S., (2003) “Feature extraction for the analysis of colon status from the endoscopic images”, *Biomed Eng Online*.
- [11] Scholkopf, B., and Smola, A., (2001) “*Learning with kernels*”, MIT Press.
- [12] Theodoridis, S., Koutroumbas, K., (2003) “*Pattern Recognition*”, Academic Press, 2nd edition, ISBN 978-0126858754.

Σχεδίαση συστολικής αρχιτεκτονικής νευρωνικού δικτύου SOM και υλοποίηση σε FPGA

Ευάγγελος Λογαράς¹

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών,
Τμήμα Πληροφορικής και Τηλεπικοινωνιών

Περίληψη Στόχος της παρούσας διπλωματικής εργασίας ήταν η περιγραφή σε γλώσσα VHDL και η παραμετροποιημένη υλοποίηση (IP core) συστολικής αρχιτεκτονικής νευρωνικού δικτύου τύπου SOM (Self-Organizing Map) σε μία μονάδα FPGA. Νευρωνικά δίκτυα αυτής της μορφής εκπαιδεύονται χωρίς επιτήρηση (unsupervised learning) και βρίσκουν εφαρμογή σε διαδικασίες κατηγοριοποίησης δεδομένων (data clustering) για τα οποία δεν υπάρχει πρότερη γνώση των στατιστικών χαρακτηριστικών τους. Τα τελευταία χρόνια οι μονάδες FPGA βρίσκουν ευρεία εφαρμογή για τη μετάβαση αλγορίθμων από λογισμικό σε υλικό. Σκοπός της δικιάς μας υλοποίησης ήταν η επιτάχυνση λειτουργίας του αλγορίθμου SOM μέσω της υλοποίησης του στο FPGA. Το σχέδιο της αρχιτεκτονικής του νευρωνικού δικτύου είναι πλήρως παραμετροποιήσιμο ως προς το πλήθος των νευρώνων που περιλαμβάνει, το πλήθος των στοιχείων των προς επεξεργασία διανυσμάτων και το πλήθος των bits αναπαράστασης των δεδομένων και των βαρών του δικτύου. Η υλοποίηση του σχεδίου έγινε με τη χρήση μονάδας FPGA τύπου Virtex-II Pro της εταιρείας Xilinx. Μέσω της σχεδίασης που έχει γίνει ενοείται η επεξεργασία διανυσμάτων μεγάλου μεγέθους με χιλιάδες στοιχεία, ενώ ο ρυθμός χρονισμού της αρχιτεκτονικής SOM μπορεί να ξεπεράσει τα 100MHz.

Λέξεις κλειδιά: FPGA, Νευρωνικά Δίκτυα, SOM, συστολική αρχιτεκτονική, VHDL

1. Εισαγωγή

Η διαδικασία εξόρυξης πληροφοριών από ένα μεγάλο πλήθος δεδομένων, αποτελεί σήμερα βασικό εργαλείο σε πολλούς τομείς της επιστήμης. Οι μέθοδοι οι οποίες χρησιμοποιούνται για το σκοπό αυτό βασίζονται κυρίως σε τομείς των μαθηματικών, όπως είναι θεωρία πιθανοτήτων, η στατιστική και η αριθμητική ανάλυση. Ο απώτερος σκοπός της επεξεργασίας ενός συνόλου δεδομένων (dataset) είναι συνήθως η εύρεση πρωτοτύπων (patterns) ή η εκτίμηση ενός μαθηματικού μοντέλου που να εκφράζει επαρκώς τα δεδομένα. Ακόμα πολλές φορές θέλουμε να διαμερίσουμε τα δεδομένα που διαθέτουμε σε ομάδες με κοινά χαρακτηριστικά. Η διαμέριση αυτή τις περισσότερες φορές γίνεται σύμφωνα με τη θέση που κατέχουν τα δεδομένα σε έναν πολυδιάστατο χώρο, που ίσως φανερώνει κοινές ιδιότητες.

¹ Επιβλέπων: Η. Μανωλάκος, Αναπλ. Καθηγητής

Μία αρκετά δημοφιλής μέθοδος για την κατηγοριοποίηση δεδομένων με μη αναλυτικό τρόπο είναι η *k*-nearest neighbor [1], κατά την οποία ένα στοιχείο κατατάσσεται σε μία συγκεκριμένη ομάδα-κατηγορία σύμφωνα με την ομάδα στην οποία κατατάσσονται και τα γειτονικά, ως προς το χώρο, σε αυτό στοιχεία. Για τον ορισμό των γειτονικών στοιχείων χρησιμοποιείται συνήθως ως μέτρο η Ευκλείδεια απόσταση του ζητούμενου στοιχείου ως προς τα υπόλοιπα. Επίσης δημοφιλής είναι και άλλη μία μέθοδος κατηγοριοποίησης, γνωστή ως *k*-means [1]. Ο αλγόριθμος *k*-means δημιουργεί περιοχές στο χώρο των δεδομένων και εντοπίζει το κέντρο των περιοχών αυτών. Τα προς επεξεργασία διανύσματα κατηγοριοποιούνται στην περιοχή από της οποίας το κέντρο απέχουν τη μικρότερη απόσταση. Η *k*-means ανήκει στην οικογένεια μεθόδων κατηγοριοποίησης που βασίζονται στον υπολογισμό μίας συνάρτησης κόστους (cost function), για τη βελτίωση της τιμής της οποίας υπάρχει κάποιος στόχος και μέσω μιας επαναληπτικής διαδικασίας βελτιστοποίησης αποφασίζεται η κατηγοριοποίηση.

Στις μεθόδους κατηγοριοποίησης κατατάσσονται επίσης και τα νευρωνικά δίκτυα, τα οποία χρησιμοποιούνται για την προσέγγιση μη γραμμικών συναρτήσεων. Η δυνατότητα των νευρωνικών δικτύων να εκπαιδεύονται ώστε να αναγνωρίζουν συμπεριφορές κάποιας μορφής (στατιστικές, χρονικές, χωρικές κ.α.) που εμφανίζονται σε ένα μεγάλο πλήθος δεδομένων, τα καθιστά πολύτιμο εργαλείο σε τέτοιες αναλύσεις. Τα νευρωνικά δίκτυα βασίζονται στην εκμάθηση (learning) των χαρακτηριστικών της κατανομής των δεδομένων που επεξεργάζονται, η οποία πραγματοποιείται μέσω της αλλαγής των βαρών των νευρώνων κατά τη διαδικασία εκπαίδευσης του δικτύου.

Η εκπαίδευση ενός νευρωνικού δικτύου μπορεί να γίνεται με επιτήρηση (supervised) ή χωρίς επιτήρηση (unsupervised). Στην πρώτη περίπτωση το νευρωνικό δίκτυο μαθαίνει τις παραμέτρους της κατανομής μέσω παραδειγμάτων, δηλαδή παρουσιάζουμε ως είσοδο στο δίκτυο τα δεδομένα τα οποία θέλουμε να επεξεργαστούμε και στην έξοδο παρέχουμε στο δίκτυο την επιθυμητή απόκριση. Το νευρωνικό δίκτυο διαμορφώνει τα βάρη του έτσι ώστε να παράγει την επιθυμητή έξοδο με το μικρότερο σφάλμα. Στην περίπτωση της εκπαίδευσης ενός νευρωνικού δικτύου χωρίς επιτήρηση, θέλουμε να αναλύσουμε δεδομένα, για τα οποία όμως δεν ξέρουμε ποια είναι η κατανομή τους και αφήνουμε μόνο του το δίκτυο να την εξάγει από τις εμφανιζόμενες συσχετίσεις μεταξύ τους. Τα δεδομένα παρουσιάζονται στο δίκτυο, χωρίς να παρέχουμε εμείς την επιθυμητή έξοδο. Η εκπαίδευση του δικτύου και η προσαρμογή των βαρών του κατ' αυτόν τον τρόπο βασίζεται στον "ανταγωνισμό" (competitive learning) μεταξύ των νευρώνων ώστε τα βάρη τους να ανταποκρίνονται στην κατανομή των δεδομένων εισόδου. Για την μέτρηση της ανταπόκρισης ενός νευρώνα χρησιμοποιείται συνήθως κάποιο κριτήριο εκτίμησης της απόστασης των βαρών του από κάθε εισερχόμενο διάνυσμα. Ο νευρώνας με τη μικρότερη απόσταση "επιβραβεύεται" (ενισχύει τα βάρη του) και μέσω αυτής της επαναληπτικής διαδικασίας τα βάρη των νευρώνων δημιουργούν ένα *n*-διάστατο χώρο, όπου *n* το πλήθος των βαρών, ο οποίος εκφράζει τα χαρακτηριστικά των δεδομένων.

1.1 Τα δίκτυα Self-Organizing Maps (SOM)

Σύμφωνα με τις αρχές της ανταγωνιστικής εκμάθησης (competitive learning) είναι η οικογένεια των νευρωνικών δικτύων που ονομάζονται SOM (Self-Organizing Map) [2]. Κάθε νευρώνας του δικτύου αποτελεί μία ομάδα στην οποία μπορούν να κατηγοριοποιηθούν τα δεδομένα και σε αντίθεση με άλλες μεθόδους η θέση μίας ομάδας

μέσα στον πολυδιάστατο χώρο των δεδομένων επηρεάζει και τη θέση των γειτονικών της ομάδων. Η εκπαίδευση των νευρώνων ομοιάζει κάπως με τον τρόπο που λειτουργούν οι νευρώνες στον ανθρώπινο εγκέφαλο, όπου σχηματίζουν περιοχές με κοινές λειτουργίες. Η εκπαίδευση ενός δικτύου SOM γίνεται σειριακά κάθε φορά που εισέρχεται ένα διάνυσμα στην είσοδο και όχι αφού παρουσιαστούν πρώτα όλα τα διανύσματα στο δίκτυο, όπως συμβαίνει στην περίπτωση του k-means. Ο τρόπος με τον οποίο τοποθετούνται οι νευρώνες μέσα στον χώρο των δεδομένων μετά το πέρας της εκπαίδευσης ενός δικτύου SOM, συμβαδίζει με τα κοινά χαρακτηριστικά που παρουσιάζουν τα δεδομένα και αποτελεί μία πολύ καλή μέθοδο για την οπτική τους αναπαράσταση (visualization).

Ένα νευρωνικό δίκτυο SOM μετασχηματίζει τον πολυδιάστατο χώρο των δεδομένων σε ένα χώρο λιγότερων διαστάσεων² στον οποίο αναδεικνύονται τα χαρακτηριστικά και οι συσχετίσεις τους (feature selection). Η λειτουργία ενός δικτύου SOM αποτελείται από δύο βασικές φάσεις: α) την εκπαίδευση (learning) του δικτύου κατά την οποία τα βάρη των συνάψεων κάθε νευρώνα τροποποιούνται ώστε ο συγκεκριμένος νευρώνας να εκπαιδευτεί και να κατηγοριοποιεί στοιχεία της εισόδου με συγκεκριμένες ιδιότητες και β) την φάση της ανάκλησης (recall) όπου νέα δεδομένα εισάγονται στο δίκτυο με στόχο την κατηγοριοποίηση τους. Και στις δύο φάσεις οι νευρώνες του δικτύου λειτουργούν ανταγωνιστικά μεταξύ τους (competitive learning), με στόχο να εντοπιστεί ο νευρώνας νικητής, δηλαδή αυτός με την μικρότερη απόσταση από το εκάστοτε διάνυσμα εισόδου. Εφόσον βρεθεί ο νευρώνας αυτός, αν είμαστε στη φάση της εκπαίδευσης θα ανανεωθούν τα βάρη του και των γειτονικών του νευρώνων, ενώ αν είμαστε στη φάση της ανάκλησης τότε το τρέχον διάνυσμα εισόδου θα κατηγοριοποιηθεί στον νευρώνα νικητή που θα έχει και την μικρότερη απόσταση από το στοιχείο της εισόδου. Η γειτονιά κάθε νευρώνα ορίζεται σύμφωνα με την απόσταση του από τους υπόλοιπους νευρώνες. Η απόσταση στο θεωρητικό μοντέλο υπολογίζεται με τη χρήση της Ευκλείδειας απόστασης.

Τα δίκτυα SOM έχουν βρει εφαρμογές σε πολλές διαδικασίες επεξεργασίας δεδομένων, κυρίως όταν ο όγκος τους είναι μεγάλος και οι διαστάσεις τους πολλές και απαιτείται κάποια εξαγωγή των βασικών χαρακτηριστικών που τα περιγράφουν. Χαρακτηριστικό παράδειγμα αποτελεί η χρήση ενός δικτύου SOM για την μείωση των χρωμάτων μίας εικόνας [4], μέσω της δημιουργίας μιας μειωμένης παλέτας χρωμάτων που θα αναδεικνύουν με τον καλύτερο τρόπο τα χαρακτηριστικά της εικόνας, ενώ παράλληλα θα συμπιέζουν το μέγεθος της. Στο άρθρο [5] ένα δίκτυο SOM χρησιμοποιείται για αναζήτηση προτύπων (pattern recognition) και για την κατηγοριοποίηση διαστημικών εικόνων στην φασματική περιοχή του ορατού και του υπέρυθρου. Στο άρθρο [6] γίνεται αναφορά στη χρήση δικτύου SOM για την εφαρμογή μεθόδου τυφλής αποσυνέλιξης³ (blind deconvolution) και την αναγνώριση των χαρακτηριστικών ενός τηλεπικοινωνιακού καναλιού μέσω της επεξεργασίας του λαμβανόμενου σήματος, ενώ στο άρθρο [7] γίνεται αναφορά στην χρήση ενός δικτύου SOM για την αναγνώριση του περιβάλλοντος μέσα στο οποίο κινείται ένα υποθαλάσσιο όχημα.

² Η διαδικασία αυτή αναφέρεται στη βιβλιογραφία ως dimensionality reduction [3].

³ Μέσω της συγκεκριμένης διαδικασίας αναζητούμε τις παραμέτρους ενός μοντέλου παρατηρώντας τα δεδομένα τα οποία παράγει ως έξοδο.

1.2 Υλοποίηση δικτύων SOM με υλικό

Αν και ο αλγόριθμος λειτουργίας του SOM είναι υπολογιστικά απλούστερος σε σύγκριση με άλλων νευρωνικών δικτύων, όπως του multilayer perceptron [3], όταν απαιτείται η επεξεργασία μεγάλου πλήθους διανυσμάτων με πολλές διαστάσεις σε πραγματικό χρόνο, τότε ακόμα και σύγχρονοι υπολογιστές αδυνατούν να αντεπεξέλθουν στις ανάγκες μιας εφαρμογής πραγματικού χρόνου (real-time), ειδικά όταν το δίκτυο αποτελείται από πολλούς νευρώνες. Σε αυτές τις περιπτώσεις κρίνεται σκόπιμη η υλοποίηση των δικτύων με υλικό (hardware) και υπάρχουν αρκετές σχετικές αναφορές στη βιβλιογραφία. Οι υλοποιήσεις αυτές γίνονται είτε με custom VLSI (Very Large Scale Integration) σχεδίαση [8], [9] και την δημιουργία ενός ASIC (Application Specific Integrated Circuit) είτε με την περιγραφή του σχεδίου σε κάποια γλώσσα HDL (Hardware Description Language) και την υλοποίηση του σε μονάδα προγραμματιζόμενης λογικής FPGA (Field Programmable Gate Array) [4],[10].

Η πλειονότητα των υλοποιήσεων SOM σε υλικό στοχεύει σε μία συγκεκριμένη εφαρμογή και η κυρίαρχη παράλληλη αρχιτεκτονική του δικτύου των νευρώνων είναι η SIMD (Single Instruction Multiple Data) [10]. Ειδικότερα για τις υλοποιήσεις σε ASIC προτιμάται η σχεδίαση με τη χρήση της τεχνολογίας standard-cell, εφόσον και η δομή ενός δικτύου SOM ευνοεί μία τέτοια προσέγγιση. Οι νευρώνες υλοποιούνται συνήθως σε μορφή πίνακα 2-D, ενώ μία μονάδα ελέγχου δίνει τις κατάλληλες εντολές στους νευρώνες και τους παρέχει τα προς επεξεργασία δεδομένα. Όπως συμβαίνει στις υλοποιήσεις σε υλικό, όπου πολύπλοκες μαθηματικά πράξεις προσεγγίζονται με πιο απλές μεθόδους, έτσι και στις περισσότερες υλοποιήσεις των SOM, η χρήση της Ευκλείδιας απόστασης που απαιτείται αντικαθίσταται από τη χρήση της απόστασης Manhattan. Στις περισσότερες υλοποιήσεις τα δεδομένα αναπαρίστανται με 8-bit, ενώ για τα βάρη επιλέγεται αναπαράσταση 12-bit, ώστε να υπάρχει μεγαλύτερη ακρίβεια στους υπολογισμούς.

Αν και η SIMD αρχιτεκτονική προσεγγίζει πολύ καλά την δομή ενός δικτύου SOM⁴, δημιουργεί πολύ μεγάλα μονοπάτια (critical paths) στο σχέδιο, εξαιτίας της ταυτόχρονης μετάδοσης των δεδομένων και των σημάτων ελέγχου προς όλους τους νευρώνες του δικτύου. Τα μονοπάτια αυτά μπορεί να καθυστερήσουν τη λειτουργία του δικτύου, ενώ δημιουργούν και πρόβλημα στη σχεδίαση, εφόσον αποτελούν γραμμές οι οποίες διατρέχουν το σχέδιο σε μεγάλο μήκος, ιδιαίτερα όταν το SOM δίκτυο αποτελείται από πολλούς νευρώνες.

2. Συστηματική σχεδίαση συστολικής αρχιτεκτονικής SOM

Το σύστημα το οποίο σχεδιάσαμε [11] για τις ανάγκες της υλοποίησης του νευρωνικού δικτύου SOM, κάνει χρήση σε μεγάλο βαθμό των δυνατοτήτων παραλληλισμού που παρέχουν οι μονάδες FPGA. Η συστολική αρχιτεκτονική την οποία επιλέξαμε είναι επιθετικά σωληνωμένη και προσπαθεί να μειώσει τα κρίσιμα μονοπάτια τα οποία εμφανίζονται σε άλλες υλοποιήσεις όπου έχει γίνει χρήση αρχιτεκτονικής SIMD.

⁴ Η αρχιτεκτονική SIMD προσεγγίζει πολύ καλά την υλοποίηση σχεδίων με πολλά επεξεργαστικά στοιχεία που λειτουργούν παράλληλα και επιτελούν την ίδια λειτουργία σε κάθε κύκλο, όπως συμβαίνει και στην περίπτωση ενός δικτύου SOM.

Επίσης το σχέδιο είναι παραμετροποιήσιμο ως προς το μέγεθος του νευρωνικού δικτύου που υλοποιείται κάθε φορά καθώς επίσης και ως προς τον όγκο των δεδομένων που μπορεί να επεξεργαστεί. Πιο συγκεκριμένα οι παράμετροι του σχεδίου είναι:

- το πλήθος των νευρώνων του δικτύου (N).
- το μέγεθος των προς επεξεργασία διανυσμάτων (M).
- η ανάλυση σε bits των στοιχείων των διανυσμάτων και των βαρών του δικτύου.

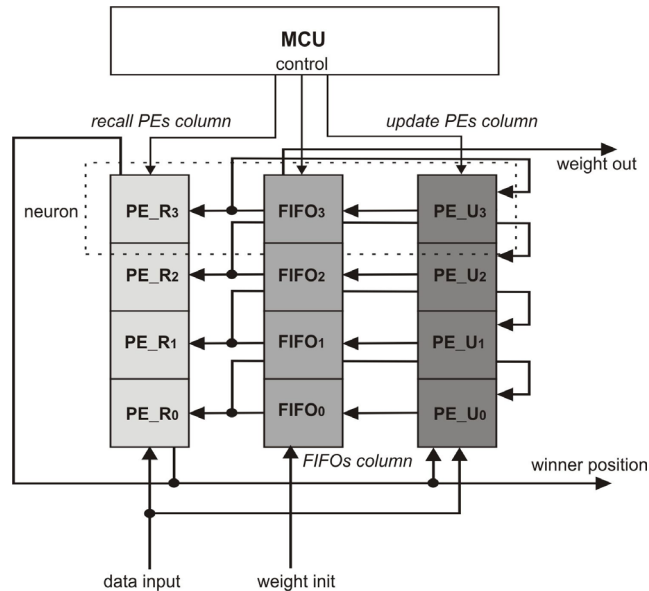
Η περιγραφή του σχεδίου ως ενός συνθέσιμου παραμετροποιήσιμου IP (Intellectual Property) core του δίνει τη δυνατότητα να προσαρμόζεται στις ανάγκες των εφαρμογών, σύμφωνα με τις απαιτήσεις που παρουσιάζει η κάθε μία ξεχωριστά. Επίσης η υλοποίηση του μπορεί να γίνει σε μονάδες FPGA με διαφορετική χωρητικότητα, εφόσον με την αλλαγή των παραμέτρων αυξομειώνεται και η πολυπλοκότητα του σχεδίου. Επιπλέον αξιοποιείται και η δυνατότητα δυναμικής αναδόμησης (dynamic reconfiguration) των σύγχρονων FPGA και με το να μπορεί πιθανά να αλλάξει δυναμικά η δομή του SOM δικτύου.

2.1 Αρχές συστολικής σχεδίασης

Οι συστολικές αρχιτεκτονικές αποτελούνται συνήθως από δίκτυα επεξεργαστικών στοιχείων (PEs – Processing Elements), τα οποία ρυθμικά εκτελούν διάφορους υπολογισμούς με τα δεδομένα τα οποία δέχονται σαν είσοδο, και προωθούν τα αποτελέσματα μέσα στο σύστημα, με τέτοιον τρόπο ώστε να υπάρχει μία συνεχής και σωληνωμένη ροή δεδομένων. Μία συστολική αρχιτεκτονική παρουσιάζει δύο πολύ σημαντικά χαρακτηριστικά τα οποία απαιτούνται στη σχεδίαση ψηφιακών συστημάτων, είτε αυτή γίνεται σε ASIC είτε με τη χρήση στοιχείων προγραμματιζόμενης λογικής, όπως στην περίπτωση μας: α) την κανονικότητα (regularity) στην λειτουργία του συστήματος και β) ότι η δομή του συστήματος διασπάται σε περισσότερες και απλούστερες δομικές μονάδες. Ειδικά το δεύτερο χαρακτηριστικό πληρούται στην περίπτωση μας και σε σχεδιαστικό επίπεδο, με τον επιμερισμό του σχεδίου σε τρεις λειτουργικές κολώνες, κάθε μία από τις οποίες αποτελείται είτε από PEs (κολώνες recall και update) είτε από στοιχεία μνήμης (κολώνα FIFO – First In First out), αλλά και στο επίπεδο της υλοποίησης, με την προσπάθεια μας για απεικόνιση του σχεδίου πάνω στις κολώνες του FPGA στις οποίες είναι διατεταγμένα τα στοιχεία λογικής (CLBs – Configurable Logic Blocks) και μνήμης (BRAMs – Block Random Access Memories).

Συνήθως όλα τα PEs σε μία συστολική αρχιτεκτονική είναι πανομοιότυπα και οι συνδέσεις μεταξύ τους πλήρως σωληνωμένες. Στο σχέδιο μας θα υπάρχουν δύο συστολικά arrays (κολώνες) με διαφορετικές δομές από PEs (PE_R και PE_U – recall και update), τα οποία θα επικοινωνούν με τοπικές μνήμες FIFO, στις οποίες θα καταχωρούνται τα βάρη των νευρώνων. Η κολώνα στοιχείων PE_R θα υλοποιεί τη διαδικασία της ανάκλησης στο δίκτυο SOM και η κολώνα PE_U θα υλοποιεί την διαδικασία της εκπαίδευσης του δικτύου και της ανανέωσης των βαρών των νευρώνων. Η συνολική τοπολογία του δικτύου (Σχήμα 1) θα αποτελείται από τρεις κολώνες και μία μονάδα ελέγχου MCU (Module Control Unit). Όλα τα αντίστοιχα PEs καθώς και οι μνήμες θα οργανωθούν στην αντίστοιχη κολώνα, έτσι ώστε να έχουμε την recall column, την update column και την FIFO column, τα αντίστοιχα στοιχεία των οποίων θα επικοινωνούν οριζόντια μεταξύ τους δημιουργώντας έτσι τη

συνολική δομή που χρειαζόμαστε για ένα νευρώνα. Ο ρόλος της MCU θα είναι να τροφοδοτεί τις κολώνες update και recall με τα δεδομένα εισόδου καθώς και να παράγει μία σειρά σημάτων για τον έλεγχο όλου του δικτύου.



Σχήμα 1 Συνολική αρχιτεκτονική δικτύου SOM ($N=4$).

Το βασικό βήμα στη σχεδίαση ήταν η αναπαράσταση των αλγορίθμων ανάκλησης και εκπαίδευσης του δικτύου SOM στα αντίστοιχα DGs (Dependence Graphs) και η απεικόνιση (προβολή) των DGs στο SFG (Signal Flow Graphs) που μοντελοποιούν τη δομή της συστολικής αρχιτεκτονικής. Το διάγραμμα εξαρτήσεων DG (Dependence Graph) για την περιγραφή του παραπάνω αλγορίθμου φαίνεται στο Σχήμα 2α., όπου εκτελείται η διαδικασία ευρέσεως του νευρώνα νικητή, για ένα δίκτυο με $M=6$ στοιχεία στο διάνυσμα εισόδου και $N=4$ νευρώνες. Το recall DG, αποτελείται από δύο ειδών nodes όπως φαίνεται και στο Σχήμα 2β, τα μαύρα, τα οποία υπολογίζουν την απόσταση Manhattan, και τα γκρίζα τα οποία εντοπίζουν τον νευρώνα νικητή. Η μορφή των δύο nodes φαίνεται στο Σχήμα 2γ. Η διαδικασία συστηματικής απεικόνισης των κόμβων του DG σε συγκεκριμένα PEs του SFG είναι η ακόλουθη [12]:

- Επικογή του διανύσματος προβολής (Projection vector) \mathbf{d} : κόμβοι του DG οι οποίοι απέχουν κατά \mathbf{d} , ή πολλαπλάσια του \mathbf{d} , θα εκτελούνται στον ίδιο επεξεργαστή (PE).

- $\mathbf{d}^T = [1 \ 0]$

Με το διάνυσμα προβολής που επιλέξαμε, όλοι οι κόμβοι που σχετίζονται με έναν συγκεκριμένο νευρώνα στο δίκτυο SOM θα απεικονιστούν στο ίδιο PE στο SFG. Συνεπώς στο SFG θα προκύψουν τόσα PEs όσοι και οι νευρώνες του δικτύου SOM.

- Επιλογή του διανύσματος βάσης του χώρου επεξεργαστών (Processor space vector) \mathbf{p} :
κόμβοι του DG με δείκτη $I^T=(i,j)$ θα εκτελούνται από το PE με δείκτη στο SFG $\mathbf{p}^T I$.

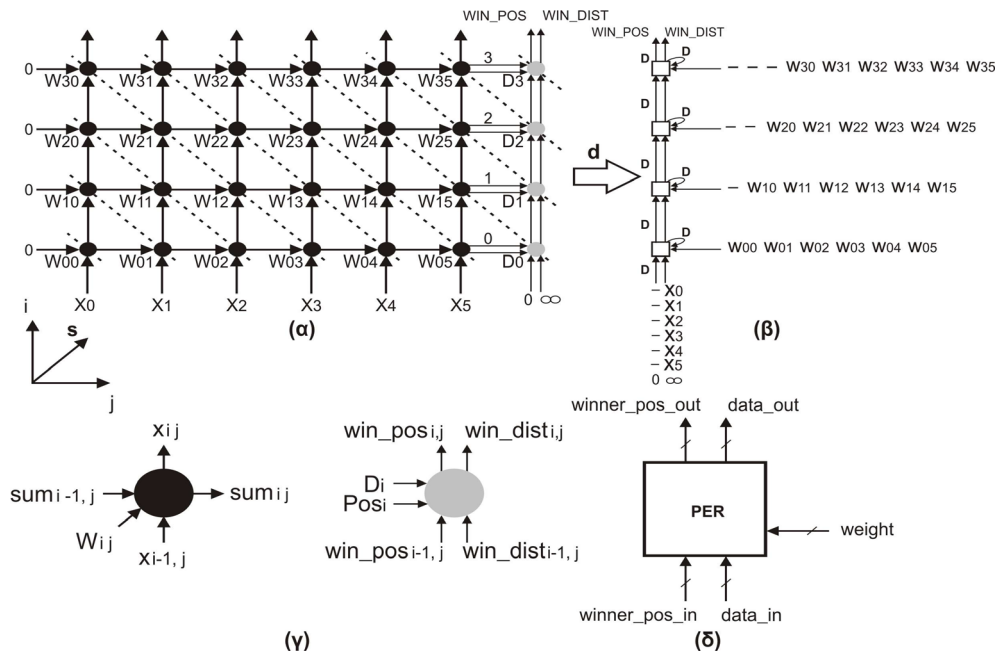
- $\mathbf{p}^T=[0 \ 1]$

Το διάνυσμα \mathbf{p} πρέπει να είναι κάθετο στο \mathbf{d} ($\mathbf{p}^T \mathbf{d} = 0$). Η συνθήκη αυτή εξασφαλίζει ότι κόμβοι του DG που απέχουν κατά \mathbf{d} , ή πολλαπλάσια του, θα εκτελούνται στον ίδιο επεξεργαστή στο SFG.

- Επιλογή του διανύσματος γραμμικού χρονοπρογραμματισμού (Scheduling vector) \mathbf{s} :
κόμβοι του DG με δείκτη $I^T=(i,j)$ θα εκτελούνται τη χρονική στιγμή $\mathbf{s}^T I$

- $\mathbf{s}^T=[1 \ 1]$

Για το scheduling vector πρέπει να ισχύει ότι $\mathbf{s}^T \mathbf{d} \neq 0$. Η συνθήκη αυτή εξασφαλίζει ότι κόμβοι οι οποίοι θα απεικονιστούν στον ίδιο επεξεργαστή δεν θα εκτελούνται την ίδια χρονική στιγμή. Η επιλογή του \mathbf{s} έγινε έτσι ώστε να υποστούν σωλήνωση τα στοιχεία της εισόδου.

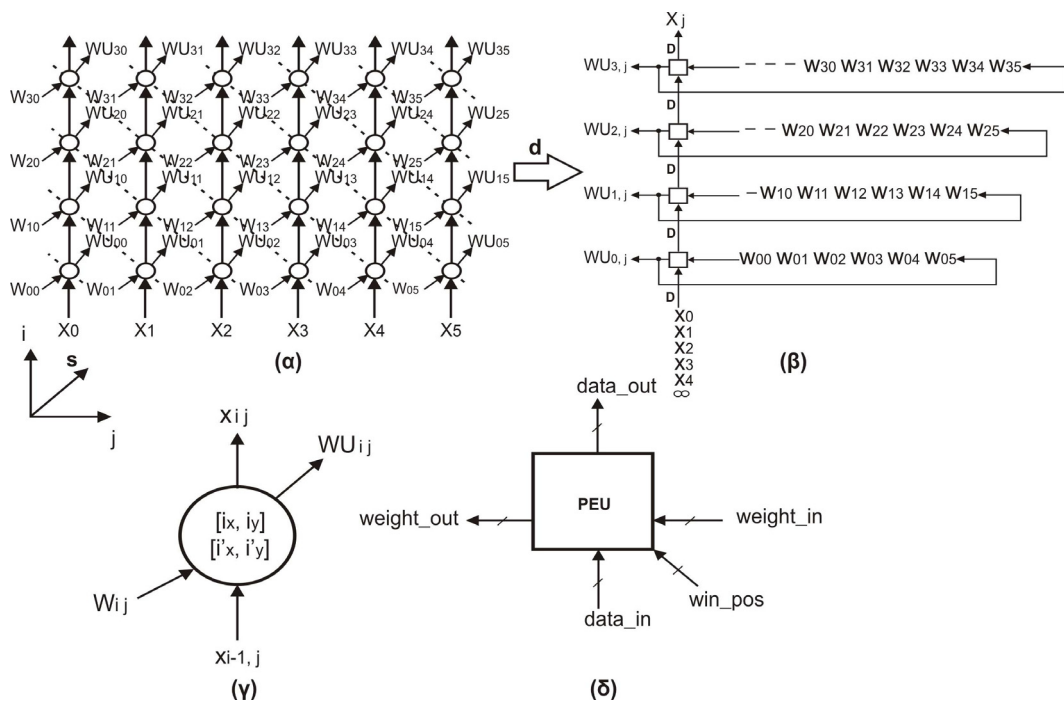


Σχήμα 2 (α) DG για την διαδικασία του recall, $N=4$, $M=6$, (β) προκύπτων SFG μετά την απεικόνιση, (γ) μάκρος (i,j) κόμβος του DG και γκρίκος κόμβος i του DG, (δ) εξωτερικές διασυνδέσεις του PE.

Τα αντίστοιχα στοιχεία σχεδίασης της κολώνας PE_U είναι τα ακόλουθα:

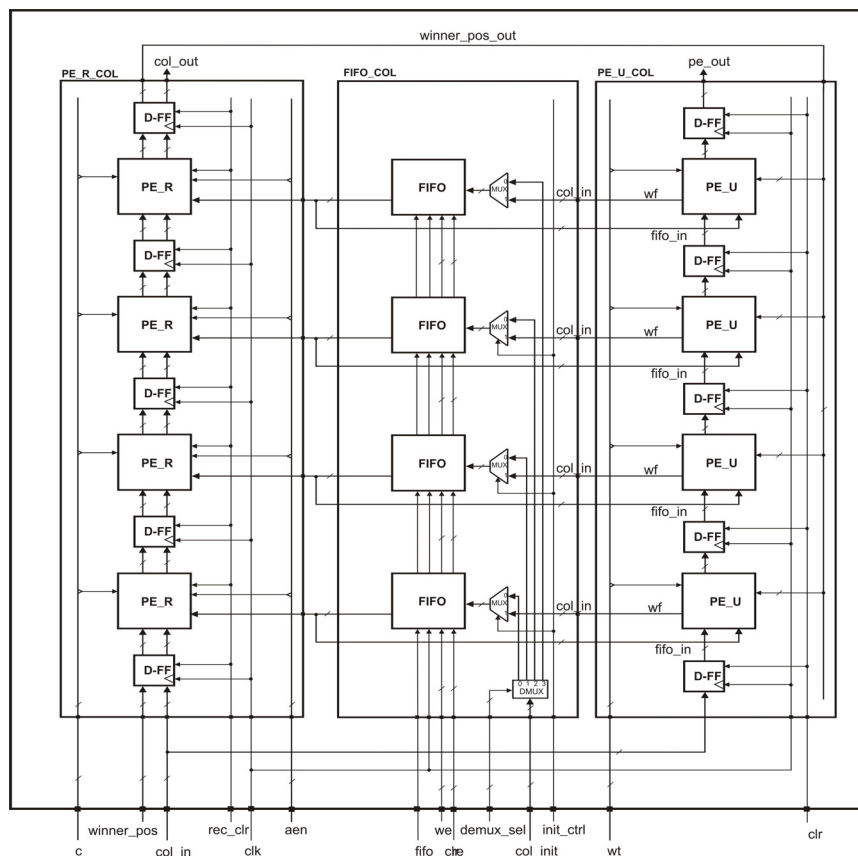
$$\begin{aligned}
 - \mathbf{d}^T &= [1 \ 0] \\
 - \mathbf{p}^T &= [0 \ 1] \\
 - \mathbf{s}^T &= [1 \ 1]
 \end{aligned}$$

όπως φαίνεται και στο Σχήμα 3.



Σχήμα 3 (α) DG για την διαδικασία ανανέωσης των βαρών, $N=4$, $M=6$, (β) προκύπτων SFG μετά την απεικόνιση, (γ) κόμβος του DG, (δ) εξωτερικές διασυνδέσεις του PE.

Το συνολικό διάγραμμα διασύνδεσης των 3 κολώνων φαίνεται στο Σχήμα 4.



Σχήμα 4 Συνολική δομή δικτύου SOM με $N=4$ νευρώνες και $M=4$ στοιχεία στο διάνυσμα εισόδου.

3. Αποτελέσματα υλοποίησης

Η παραμετροποιήσιμη αρχιτεκτονική δικτύου SOM που σχεδιάστηκε σε αυτήν την εργασία έχει περιγραφεί πλήρως σε VHDL (Very high speed integrated circuit Hardware Description Language) και η σύνθεση του παραγόμενου υλικού έγινε με τη χρήση του synthesizer XST της Xilinx. Το σχέδιο μπορεί να υλοποιηθεί σε οποιοδήποτε FPGA της Xilinx που περιέχει εσωτερικές block RAMs (BRAMs), οι οποίες χρειάζονται για την υλοποίηση των μνημών FIFO, όπου καταχωρούνται τα βάρη των νευρώνων.

Ο μέγιστος αριθμός νευρώνων που μπορούν να χωρέσουν σε ένα FPGA, εξαρτάται από το πλήθος των CLBs που περιέχει η συγκεκριμένη μονάδα. Λαμβάνοντας υπ' όψιν το γεγονός, ότι το 25% των CLBs θα δεσμευθεί για την διεπαφή με κάποια εξωτερική μνήμη δεδομένων ή κάποιον ελεγκτή για πιθανές διαδικασίες reconfiguration του FPGA, υποθέσαμε ότι το δίκτυο SOM δεν πρέπει να καταλαμβάνει πάνω από το 75% των διαθέσιμων CLBs. Για δεδομένες τιμές ανάλυσης για τα βάρη στα 8-bit και για τα δεδομένα

στα 12-bit (τυπικές τιμές που χρησιμοποιούνται και στο [9]), στον Πίνακα 1 αναφέρεται το πλήθος των στοιχείων M_{max} στα διανύσματα εισόδου το οποίο μπορεί να υποστηριχθεί για υλοποιήσεις σε FPGA της Xilinx των οικογενειών Virtex-II και Virtex-II Pro. Για κάθε υλοποίηση ο πίνακας επίσης αναφέρει την μέγιστη συχνότητα λειτουργίας, την αναμενόμενη απόδοση στις δύο φάσεις λειτουργίας του δικτύου (P_r - recall και P_l - learning) μετρημένες σε MCPS (Mega Connections Per Second) και MCUPS (Mega Connection Updates Per Second) αντίστοιχα, καθώς επίσης αναφέρει και τα ποσοστά κάλυψης στο FPGA των CLBs και των BRAMs.

| FPGA | N_{maz} | M_{max} | f (MHz) | P_r (MCPS) | P_l (MCUPS) | CLBs % | BRAMs % |
|------------|-----------|-----------|---------|--------------|---------------|--------|---------|
| XC2V2000-6 | 25 | 4096 | 150 | 3750 | 1794 | 73 | 90 |
| XC2V4000-6 | 56 | 4096 | 150 | 8400 | 2899 | 75 | 94 |
| XC2V6000-6 | 100 | 2048 | 148 | 14800 | 3467 | 72 | 70 |
| XC2V8000-5 | 140 | 2048 | 127 | 17780 | 3192 | 72 | 84 |
| XC2VP30-6 | 32 | 4096 | 165 | 5280 | 2054 | 72 | 47 |
| XC2VP50-6 | 56 | 4096 | 166 | 9296 | 3208 | 72 | 48 |

Πίνακας 1 Αποτελέσματα σύνθεσης του σχεδίου για FGPA's Virtex-II και Virtex-II Pro (βάση: 8-bit, data: 12-bit).

Παρατηρούμε ότι για όλες τις υλοποιήσεις, το δίκτυο μπορεί να επεξεργαστεί εισερχόμενα διανύσματα μεγάλου μεγέθους ($M = 2K$ ή $4K$), ενώ αντίστοιχα και η απόδοση που επιτυγχάνεται είναι αρκετά υψηλή. Ενδιαφέρον παρουσιάζει η σύγκριση της παρούσας υλοποίησης με τη SIMD υλοποίηση που παρουσιάζεται στο [13]. Συγκρίνοντας τις υλοποιήσεις των δύο σχεδίων για το FPGA Virtex-II XC2V6000, παρατηρούμε ότι το δικό μας σχέδιο φαίνεται να παρουσιάζει μεγαλύτερη ευελιξία στη διαχείριση της μνήμης, καθώς μπορεί να υποστηρίξει την επεξεργασία διανυσμάτων με μέγεθος M , όσο αυτό απαιτείται κάθε φορά από την εφαρμογή. Τα δύο συστήματα επιτυγχάνουν σχεδόν τον ίδιο ρυθμό εκπαίδευσης (MCUPS), αλλά ενώ το σχέδιο στο [13] ευνοεί την υλοποίηση μεγάλων δικτύων με πολλούς νευρώνες ($N > 100$), η δικιά μας υλοποίηση ευνοεί το μεγάλο μέγεθος ($M > 1K$) στα προς επεξεργασία διανύσματα. Για κοινό πλήθος νευρώνων ($N=100$), η δικιά μας συστολική αρχιτεκτονική επιτυγχάνει καλύτερη απόδοση κατά τη διαδικασία του recall. Παρόλα αυτά, η SIMD αρχιτεκτονική που προτείνεται στο [13] μπορεί να υποστηρίξει δίκτυα με $N=288$ νευρώνες, αλλά με μειωμένο M σε σχέση με τη δικιά μας. Αυτό οφείλεται στο γεγονός ότι κάθε PE του δικτύου στη συστολική υλοποίηση πραγματοποιεί ένα μεγάλο μέρος των λειτουργιών, για τις οποίες στην SIMD υλοποίηση είναι επιφορτισμένος ο ελεγκτής του δικτύου (π.χ. ο υπολογισμός του ρυθμού εκπαίδευσης α στο PEU). Επίσης η παρουσία στο σχέδιο μας $O(N)$ συγκριτών για τον εντοπισμό του νευρώνα νικητή επιταχύνει την συγκεκριμένη διαδικασία, αλλά καταναλώνει περισσότερα CLBs σε σχέση με το σχέδιο στο [13], όπου γίνεται χρήση δέντρου συγκριτών. Αυτό σημαίνει ότι η συστολική υλοποίηση υποστηρίζει λιγότερους νευρώνες για συγκεκριμένο FPGA.

3.1 Συμπεράσματα

Με το σύστημα που σχεδιάσαμε προσπαθήσαμε να δημιουργήσουμε μια ευέλικτη και συνθέσιμη αρχιτεκτονική ενός νευρωνικού δικτύου SOM, η οποία θα μπορεί να προσαρμόζεται στις ανάγκες διαφορετικών εφαρμογών και να υλοποιείται σε ένα μόνο FPGA. Το βασικό στοιχείο για αυτήν την δυνατότητα που παρέχει το σχέδιο μας, είναι η παραμετροποίηση ως προς: α) το πλήθος των νευρώνων (N) στο SOM δίκτυο, β) το πλήθος των στοιχείων (M) σε κάθε διάνυσμα δεδομένων που εισέρχεται στο δίκτυο και γ) ο αριθμός των bits των στοιχείων εισόδου και των βαρών ξεχωριστά. Εκτός από την πλήρη παραμετροποίηση του σχεδίου, ένας άλλος στόχος ήταν η γρήγορη λειτουργία του δικτύου και η σύγκριση της ταχύτητας επεξεργασίας των εισερχόμενων διανυσμάτων με άλλες υλοποιήσεις νευρωνικών δικτύων SOM που έχουν γίνει σε FPGA. Η συστολική αρχιτεκτονική μας επέτρεψε να εφαρμόσουμε ευρείας κλίμακας σωλήνωση στο σχέδιο μας, κυρίως ανάμεσα στους διαύλους δεδομένων που διατρέχουν τους νευρώνες του δικτύου. Επίσης με τη συγκεκριμένη επιλογή μας αποφύγαμε τη δημιουργία δέντρου συγκριτών, όπως στο [9] το οποίο απαιτείται για τον εντοπισμό της μικρότερης απόστασης Manhattan κατά τη διαδικασία του recall και του update και το οποίο θα εισήγαγε καθυστερήσεις στη λειτουργία του σχεδίου.

Αναφορές

- [1] S. Theodoridis, K. Koutroumbas, *Pattern Recognition*, Academic Press, 2006.
- [2] T. Kohonen, *Self-Organizing Maps*, Springer-Verlag, 1995.
- [3] S. Haykin, *Neural Networks, a comprehensive foundation*, Prentice Hall, 1999.
- [4] N. Sudha, T. Srikanthan, B. Mailachalam, "A VLSI architecture for 3-D self-organizing map based color quantization and its FPGA implementation", Nanyang Technological University, Singapore, *Journals of Systems Architecture* 48, 2003.
- [5] T. Roush, R. Hogan, "Hardware Automated Classification of Visible and Near-Infrared Spectra Using Self-Organizing Maps", *Proc. Aerospace Conference*, 2007, pp. 1-10.
- [6] E. Baccarelli, S. Galli, "A New Approach Based on Soft Statistics to the Nonlinear Blind-Deconvolution of Unknown Data Channels", *IEEE Transactions on Signal Processing*, VOL 49, NO. 7, July 2001.
- [7] S. Nishida, K. Ishii, T. Ura, "Adaptive learning to environment using Self-Organizing Map and its application for underwater vehicles", *Proc. International Symposium on Underwater Technology (UT)*, 2004, pp. 223-228.
- [8] M. Porrmann, U. Witkowski, and U. Rückert, "A Massively Parallel Architecture for Self-Organizing Feature Maps", *IEEE Transactions on Neural Networks*, VOL 14, NO. 5,

Sept.2003.

[9] D. C. Hendry, A. Duncan, N. Lightowler, "IP Core Implementation of a Self-Organizing Neural Network", IEEE Transactions on Neural Networks, VOL 14, NO. 5, Sept. 2003.

[10] M. Porrman, M. Franzmeier, H. Kalte, U. Witkowski, U. Rückert, "A Reconfigurable SOM Hardware Accelerator", University of Paderborn, Germany, Proc. European Symposium on Artificial Neural Networks (ESANN), 2002, pp. 337-342.

[11] I. Manolakos, E. Logaras, "High Throughput Systolic SOM IP Core for FPGAs", Proc. IEEE Int'l Conf. Acoustics, Speech, and Signal Processing (ICASSP), 2007, vol. 2, pp. II-61-II-64.

[12] Keshab K. Parhi, *VLSI Signal Processing Systems – Design and Implementations*, Willey Inter-Science, 1999.

[13] M. Porrman, H. Kalte, U. Witkowski, J. C. Niemann, U. Rückert, "A Dynamically Reconfigurable Hardware Accelerator for Self-Organizing Feature Maps", Proc. World Multi-Conference on Systemics, Cybernetics and Informatics (SCI),2001, Vol. 2, pp.242-247.

Μοντελοποίηση της Συμπεριφοράς και Προσομοίωση Ασυρμάτου Δικτύου Αισθητήρων Θερμοκρασίας κατά την Εξέλιξη Δασικής Πυρκαγιάς

Δημήτριος Μανανάκης¹

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών, Τμήμα Πληροφορικής και
Τηλεπικοινωνιών, Πανεπιστημιούπολη, Ιλίσια, 15784, Αθήνα, Ελλάδα
dmanatak@di.uoa.gr

Στα πλαίσια αυτής της διπλωματικής εργασίας αναπτύξαμε έναν προσομοιωτή ο οποίος προσεγγίζει την συμπεριφορά ενός ασυρμάτου δικτύου αισθητήρων (WSN) θερμοκρασίας κατά την εξέλιξη μιας δασικής πυρκαγιάς. Ο προσομοιωτής μας είναι βασισμένος σε μια καινοτόμο προσέγγιση επεξεργασίας σήματος κατά την οποία το θερμοκρασιακό πεδίο που αισθάνεται ο κάθε αισθητήρας λόγω της εξάπλωσης του μετώπου της φωτιάς μοντελοποιείται από την μίξη K διδιάστατων κανονικών κατανομών (όπου K ένας μικρός ακέραιος). Ο προσομοιωτής αυτός, μας επιτρέπει να παράγουμε in-silico ρεαλιστικά συνθετικά δεδομένα τα οποία είναι πολύ χρήσιμα κατά την μελέτη αλγορίθμων ανίχνευσης, παρακολούθησης και πρόβλεψης της κίνησης τέτοιων καταστροφικών φαινομένων.

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Διάδοση πυρκαγιάς, Δασικές Πυρκαγιές, προσομοίωση, Θερμοκρασιακό πεδίο, δυσδιάστατη κανονική κατανομή, Mahalanobis απόσταση, αλγόριθμοι θερμοκρασιακής εκτίμησης, μηχανή εξάπλωση φωτιάς (FSE)

1. Εισαγωγή

Οι τεράστιες οικολογικές καταστροφές που έχουν προκληθεί τα τελευταία χρόνια από τις δασικές πυρκαγιές στη χώρα μας, αλλά και σε πολλά άλλα σημεία του πλανήτη, είναι σε όλους μας γνωστές. Η αύξηση της θερμοκρασίας του πλανήτη, λόγω του φαινομένου του θερμοκηπίου, καθώς και η ξηρασία που αυτή προκαλεί σε πολλές περιοχές του κόσμου δημιουργούν ευνοϊκές συνθήκες για την ανάπτυξη πυρκαγιών. Είναι πλέον καιρίο μέλημα πολλών κρατών το να βρεθούν τρόποι έγκαιρης αντιμετώπισης του καταστροφικού αυτού φαινομένου.

Ένα από τα σημαντικότερα εμπόδια που παρουσιάζονται κατά την αντιμετώπιση μιας δασικής πυρκαγιάς, είναι η αδυναμία πρόβλεψης της κίνησης του φαινομένου, δηλαδή του που θα βρίσκεται το μέτωπο της πυρκαγιάς μετά τις πρώτες 1-3 ώρες από την εκκίνησή της. Η δυνατότητα έγκαιρης και αξιόπιστης πρόβλεψης των μελλοντικών θέσεων του μετώπου μιας πυρκαγιάς αποτελεί πολύ χρήσιμη πληροφορία για την οργάνωση των επιχειρήσεων αντιμετώπισης και διαχείρισης των επιπτώσεων του καταστροφικού αυτού φαινομένου.

¹ Επιβλέπων καθηγητής: Η. Μανωλάκος

Στις μέρες μας γίνονται σημαντικές προσπάθειες για την υλοποίηση κατανεμημένων συστημάτων περιβαλλοντικής παρακολούθησης με στόχο την αντιμετώπιση των συνεπειών φυσικών καταστροφικών φαινομένων, όπως οι φωτιές, πλημμύρες κτλ. Τα *ασύρματα δίκτυα αισθητήρων* (Wireless Sensor Networks), αποτελούν τεχνολογία αιχμής για την υλοποίηση τέτοιων συστημάτων. Στα πλαίσια της διπλωματικής αυτής εργασίας, μελετήθηκε και αναπτύχθηκε ένας πρωτότυπος προσομοιωτής της συμπεριφοράς ασύρματου δικτύου αισθητήρων θερμοκρασίας που βρίσκεται τοποθετημένο στο πεδίο εξέλιξης μιας πυρκαγιάς. Ο προσομοιωτής μας, λαμβάνει ως είσοδο τη πληροφορία του μετώπου της φωτιάς, όπως αυτό παράγεται από ένα λογισμικό πακέτο πρόβλεψης της εξάπλωσης της φωτιάς (Fire Spread Engine – FSE, της ισπανικής εταιρείας Technoma) και παράγει ως έξοδο το θερμοκρασιακό πεδίο που αναμένεται να αναπτυχθεί γύρω από αυτό. Η λειτουργία του στηρίζεται σε καινοτόμα μοντελοποίηση που αναπτύξαμε και με βάση την οποία η θερμοκρασία που αντιλαμβάνεται ένας αισθητήρας θερμοκρασίας στη περιοχή της πυρκαγιάς μπορεί να προσεγγιστεί από τη μείξη K δυοδιάστατων κανονικών κατανομών (Gauss), μία κατανομή για κάθε ένα από τα K «κοντινότερα» σημεία του μετώπου ως προς τον αισθητήρα. Το K είναι ένας μικρός ακέραιος αριθμός που καθορίζει ο χρήστης και οι παράμετροι των επιμέρους κανονικών κατανομών εκτιμούνται δυναμικά με βάση φυσικές παραμέτρους (μήκος φλόγας, κατεύθυνση και ένταση του ανέμου), οι οποίες μπορεί να διαφέρουν αρκετά στα διαφορετικά σημεία του μετώπου. Για το λόγο αυτό χρησιμοποιείται η Mahalanobis και όχι η Ευκλείδεια απόσταση μεταξύ σημείων του μετώπου και αισθητήρων.

Ο προσομοιωτής που αναπτύξαμε (σε γλώσσα Matlab), είναι ο πρώτος στο είδος του και συνεισφέρει στην δημιουργία και οπτικοποίηση συνθετικών δεδομένων χώρο-χρονικής εξέλιξης του θερμοκρασιακού πεδίου (ένα είδος «θερμοκρασιακού video»), κάτω από διαφορετικές συνθήκες διάδοσης της φωτιάς. Λόγω της δυσκολίας συλλογής πραγματικών δεδομένων θερμοκρασίας σε ευρεία κλίμακα υπό πραγματικές συνθήκες, η χρήση προσομοίωσης είναι απαραίτητη για την αξιολόγηση αλγορίθμων εντοπισμού και εξέλιξης πυρκαγιών με WSN. Επιπλέον, η ύπαρξη μοντέλων πρόβλεψης του θερμοκρασιακού πεδίου επιτρέπει, κατόπιν σύγκρισης με πραγματικές τιμές θερμοκρασίας που μετρούν οι πραγματικοί κατανεμημένοι αισθητήρες στο πεδίο, την βελτίωση της ακρίβειας πρόβλεψης της μελλοντικής θέσης του μετώπου, ακόμα κι αν οι συνθήκες περιβάλλοντος μεταβάλλονται με γρήγορο ρυθμό, όπως ακριβώς συμβαίνει στις πιο καταστροφικές πυρκαγιές.

1.1. Προκλήσεις που αντιμετωπίσαμε για την ανάπτυξη του προσομοιωτή

Κατά την διάρκεια της μελέτη μας για την ανάπτυξη του προσομοιωτή συμπεριφοράς του WSN θερμοκρασίας κατά την εξέλιξη δασικής πυρκαγιάς, συναντήσαμε προκλήσεις σε αρκετά σημεία, οι οποίες έπρεπε να αντιμετωπιστούν για να μπορέσουμε να υλοποιήσουμε το βασικό στόχο της διπλωματικής.

Μια από τις σημαντικότερες προκλήσεις που έπρεπε να αντιμετωπίσουμε ήταν το πώς το κινούμενο μέτωπο μιας πυρκαγιάς επηρεάζει το θερμοκρασιακό πεδίο στην περιοχή του μετώπου. Τα φαινόμενα των δασικών πυρκαγιών είναι εξαρτώμενα από έναν μεγάλο αριθμό παραμέτρων οι οποίες μεταβάλλονται δυναμικά με το χρόνο και επομένως είναι πολύ δύσκολο να πετύχουμε μια λεπτομερή περιγραφή της

συμπεριφοράς τους. Επίσης, πολλοί παράγοντες, όπως το τρεμόπαιγμα της φλόγας, τα ζεστά ρεύματα αέρα οι αλλαγές της διεύθυνσης και της ταχύτητας του ανέμου, καθιστούν την μοντελοποίηση του θερμοκρασιακού πεδίου πολύ δύσκολη. Στην βιβλιογραφία που μελετήσαμε δεν συναντήσαμε κανένα φυσικό ή υπολογιστικό μοντέλο το οποίο να περιγράφει το θερμοκρασιακό πεδίο που δημιουργείται γύρω από το μέτωπο μιας φωτιάς. Μάλιστα, η περισσότερη βιβλιογραφία δεν χρησιμοποιεί καν μονάδες θερμοκρασίας για να περιγράψει την συμπεριφορά της φωτιάς. Γι αυτό το λόγο έπρεπε να συνδυάσουμε την διαθέσιμη γνώση που αποκομίσαμε από την βιβλιογραφία και να αναπτύξουμε ένα μοντέλο, το οποίο θα περιέγραφε το θερμοκρασιακό πεδίο γύρω από το μέτωπο μιας εξελισσόμενης πυρκαγιάς. Κύριος σκοπός της μοντελοποίησης μας ήταν να αναπτύξουμε ένα μοντέλο το οποίο θα λάμβανε ως παραμέτρους εισόδου, μεγέθη τα οποία θα μπορούσαν να εκτιμηθούν εύκολα κατά την εξέλιξη ενός φαινομένου μιας δασικής πυρκαγιάς.

Άλλη μια σημαντική πρόκληση που παρουσιάστηκε ήταν η μοντελοποίηση της κίνησης του μετώπου της πυρκαγιάς. Το σχήμα του μετώπου μιας πυρκαγιάς κατά την χρονική εξέλιξη της εξαρτάται από πολλές παραμέτρους όπως είναι, η μορφολογία του εδάφους, ο τύπος της καύσιμης ύλης, η υγρασία, η ταχύτητα και η διεύθυνση του ανέμου κτλ. Επομένως γίνεται προφανές ότι η μοντελοποίηση της χώρο-χρονικής εξέλιξης της φωτιάς αποτελεί μια πολύπλοκη διαδικασία και η προσέγγισή της είναι μια αρκετά δύσκολη υπόθεση.

Για τη μοντελοποίηση της χώρο-χρονικής εξέλιξης της φωτιάς, έχουν προταθεί κατά καιρούς διάφορα μοντέλα [1,2] τα οποία λαμβάνουν ως είσοδο διάφορες παραμέτρους του πεδίου της φωτιάς (όπως μορφολογία του εδάφους, ο τύπος της καύσιμης ύλης, η υγρασία, η ταχύτητα και η διεύθυνση του ανέμου κτλ) και εξάγουν πληροφορίες (θέση, μήκος φλόγας, ένταση της φωτιάς κτλ) για την εξέλιξη του μετώπου της φωτιάς εντός ενός πεπερασμένου διαστήματος. Ορισμένα από αυτά τα μοντέλα έχουν προγραμματιστεί και χρησιμοποιούνται σήμερα για πρόβλεψη της εξέλιξης του φαινομένου.

Οι παραπάνω προκλήσεις μας οδήγησαν σε ερωτήματα τα οποία έπρεπε να απαντηθούν προκειμένου να μπορέσουμε να μοντελοποιήσουμε και να υλοποιήσουμε τη συμπεριφορά ενός ασυρμάτου δικτύου αισθητήρων θερμοκρασίας κατά την εξέλιξη μιας δασικής πυρκαγιάς.

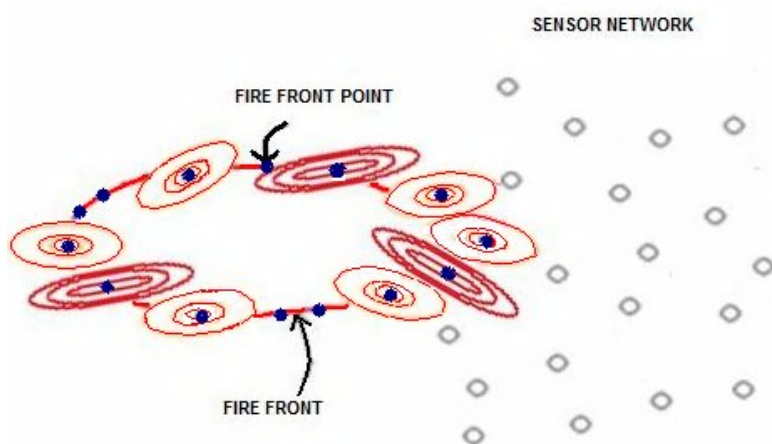
2. Αρχή Λειτουργίας Του Προσομοιωτή Θερμοκρασιακού Πεδίου

2.1. Τρόπος λειτουργίας του προσομοιωτή μας

Ο πρωτότυπος προσομοιωτής ασυρμάτων δικτύων αισθητήρων που αναπτύξαμε λαμβάνει σαν είσοδο τις θέσεις του μετώπου της πυρκαγιάς που εκτιμώνται από ένα πρόγραμμα εξέλιξης πυρκαγιάς (στην περίπτωση μας η Fire Spread Engine της TECHNOMA [3]) και παράγει σαν έξοδο τις θερμοκρασίες συναρτήση του χρόνου που καταγράφουν οι αισθητήρες που βρίσκονται τοποθετημένοι στην περιοχή. Η λειτουργία του προσομοιωτή μας είναι βασισμένη σε δύο καινοτόμα στοιχεία: (α) Το θερμοκρασιακό πεδίο που παράγεται τοπικά σε κάθε σημείο του μετώπου μοντελοποιείται με τη χρήση δισδιάστατων κανονικών κατανομών με πλήρες μητρώο συνδιασποράς το οποίο ανανεώνεται σε κάθε βήμα της προσομοίωσης και εξαρτάται

από παραμέτρους όπως το μήκος της φλόγας και οι συνθήκες ανέμου που αντιστοιχούν σε κάθε σημείο του μετώπου. (β) Η θερμοκρασία που αντιλαμβάνεται ο κάθε αισθητήρας παράγεται από μια μίξη K (όπου K μικρός ακέραιος) δισδιάστατων κανονικών κατανομών. Μια κανονική κατανομή για κάθε ένα από τα K κοντινότερα (ως προς τη Mahalanobis απόσταση) σημεία του μετώπου ως προς τον αισθητήρα.

Στο σχήμα 1 παρατηρούμε το πώς οι δισδιάστατες κατανομές τοποθετούνται πάνω στα σημεία του μετώπου. Αξίζει να σημειωθεί ότι οι δισδιάστατες κατανομές που τοποθετούνται στα διαφορετικά σημεία του μετώπου μπορούν να έχουν διαφορετικό μητρώο συνδιασποράς το οποίο εξαρτάται από τις επικρατούσες συνθήκες (μήκος φλόγας και συνθήκες ανέμου) του κάθε σημείου.



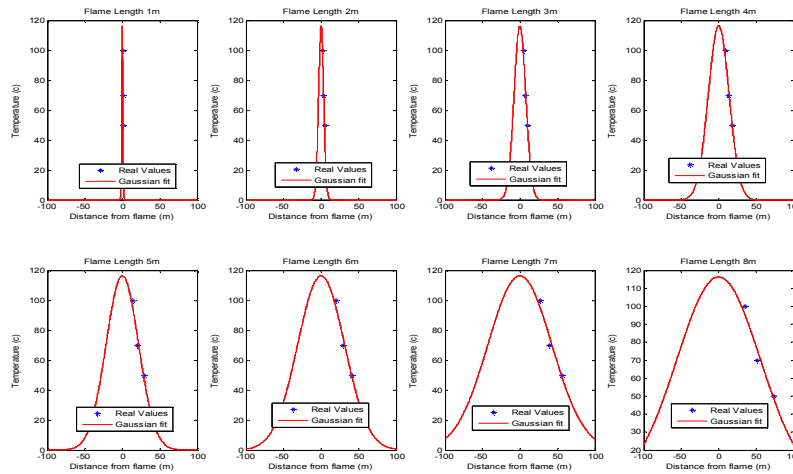
Σχήμα 1: Κατανομή των δισδιάστατων κατανομών πάνω σε ένα εξελισσόμενο μέτωπο πυρκαγιάς. Οι ομόκεντρες καμπύλες των ελλείψεων αντιστοιχίζονται σε διαφορετικές τιμές θερμοκρασιών

2.2. Μοντελοποίησης του θερμοκρασιακού πεδίου συναρτήσει της απόστασης και του ύψους της φλόγας

Όπως αναφέρθηκε, στη διαθέσιμη βιβλιογραφία δεν υπάρχουν μοντέλα τα οποία να περιγράφουν το θερμοκρασιακό πεδίο που θα αναπτυχθεί σε συγκεκριμένη απόσταση γύρω από μια φλόγα συγκεκριμένου μεγέθους. Στις περισσότερες εργασίες, η περιγραφή των φάσεων της φωτιάς γίνεται με μεγέθη όπως η θερμική ισχύς (heat flux kW/m^2). Η χρήση αυτού του φυσικού μεγέθους προτιμάται του μεγέθους της θερμοκρασίας διότι περιγράφονται καλύτερα (ενεργειακά) οι φάσεις της φωτιάς από την ανάφλεξη μέχρι την εξάλειψή της. Παρόλα αυτά οι αισθητήρες που χρησιμοποιούν οι ασύρματοι κόμβοι είναι αισθητήρες θερμοκρασίας που μετρούν

θερμοκρασίες μέχρι 120 °C. Σύμφωνα με τα αποτελέσματα που παρουσιάζονται στο [4] και [5] για την περιοχή που μετρούν (έως 120 °C) οι αισθητήρες μας, μπορούμε να θεωρήσουμε μια γραμμική σχέση μεταξύ θερμοκρασίας και θερμικής ισχύος. Θεωρώντας λοιπόν αυτή τη γραμμική σχέση, είμαστε σε θέση να εκμεταλλευτούμε από τη διαθέσιμη βιβλιογραφία όλη την γνώση των μοντέλων που περιγράφουν τη συμπεριφορά της φωτιάς με μονάδες θερμικής ισχύος.

Η μοντελοποίηση του θερμοκρασιακού πεδίου με την βοήθεια των δισδιάστατων κατανομών, στηρίζεται σε δύο μοντέλα [6] και [7]. Το πρώτο μοντέλο χρησιμοποιείται για να περιγράψει το θερμοκρασιακό πεδίο που παράγεται από μικρές φωτιές επιφανείας και μεσαίου μεγέθους φωτιές κόμης (φωτιές μήκους μικρότερες των 8m) και το δεύτερο μοντέλο χρησιμοποιείται για να περιγράψει το θερμοκρασιακό πεδίο που παράγεται από μεγάλες φωτιές (μεγαλύτερες από 8m). Τα αποτελέσματα που εξαγάγαμε, μας δείχνουν ότι η κανονικοποιημένη μονοδιάστατη κανονική κατανομή περιγράφει πολύ καλά το θερμοκρασιακό πεδίο στην κλίμακα ενδιαφέροντος (έως 120 °C) συναρτήση του μεγέθους φλόγας και της απόστασης έτσι όπως αυτό περιγράφεται από τα δύο μοντέλα. Στο σχήμα 2 φαίνονται ενδεικτικά ορισμένα από τα αποτελέσματα του ταιριάσματος των μονοδιάστατων κανονικών κατανομών με τις τιμές που προβλέπει το μοντέλο [6]. Αντίστοιχα είναι και τα ταιριάσματα που πετυχαίνουν οι μονοδιάστατες κανονικές κατανομές στις τιμές που προβλέπει το μοντέλο [7].



Σχήμα 2: Παρουσιάζονται τα ταιριάσματα (model fit) που κάνουν οι ML-εκτιμώμενες κανονικοποιημένες κανονικές κατανομές, με τις τιμές (μπλε αστερίσκοι) που δίνονται από το μοντέλο [6] για τα μικρά μήκη φλόγας (1m έως 8m).

Για να περάσουμε από τη μονοδιάστατη κανονική κατανομή στη δισδιάστατη κανονική κατανομή που μοντελοποιεί το θερμοκρασιακό πεδίο γύρω από ένα σημείο του μετώπου της φωτιάς, χρειάστηκε να βρούμε ένα τρόπο να υπολογίσουμε το

μητρώο συνδιασποράς. Ο υπολογισμός του μητρώου συνδιασποράς είναι βασισμένος σε μια εργασία του Anderson [8] σύμφωνα με την οποία αν γνωρίζουμε την ταχύτητα του ανέμου σε μια περιοχή μπορούμε να βρούμε το σχήμα του ελλειψοειδούς (το λόγο των κυρίων αξόνων της έλλειψης) που θα σχηματίσει το μέτωπο της φωτιάς τις πρώτες χρονικές στιγμές της εξέλιξής της. Αυτή η παραδοχή έχει γίνει έχοντας υπόψη ότι η ανάφλεξη της καύσιμης ύλης έχει άμεση σχέση με τη θερμική ακτινοβολία [9] και εκείνη με την σειρά της με την θερμοκρασία. Έτσι λοιπόν μπορούμε να συμπεράνουμε ότι το ελλειψοειδές σχήμα που σχηματίζει το μέτωπο της φωτιάς τις πρώτες χρονικές στιγμές έχει τον ίδιο λόγο κυρίων αξόνων με το ελλειψοειδές που σχηματίζει η θερμοκρασία γύρω από μια σημειακή εστία φωτιάς. Με βάση λοιπόν αυτή την παραδοχή μπορούμε να υπολογίσουμε από το [8] τα διαγώνια μητρώα συνδιασποράς των δυσδιάστατων κανονικών κατανομών που εφαρμόζονται στα σημεία του μετώπου. Τα πλήρη μητρώα συνδιασποράς υπολογίζονται αν γνωρίζουμε και την διεύθυνση του ανέμου σε κάθε σημείο του μετώπου που τοποθετούνται οι δισδιάστατες κανονικές κατανομές στοιχείο το οποίο μας βοηθά να στρέψουμε κατάλληλα (στην διεύθυνση του ανέμου) τον κύριο άξονα της έλλειψης.

Έτσι λοιπόν για να μοντελοποιήσουμε το θερμοκρασιακό πεδίο γύρω από τα σημεία του μετώπου της φωτιάς που μας δίνονται από ένα πρόγραμμα εξάπλωσης (Farsite [10], FSE etc) πρέπει να έχουμε διαθέσιμες τις ακόλουθες πληροφορίες.

- Τις συντεταγμένες των σημείων του μετώπου της φωτιάς.
- Μια εκτίμηση του μήκους (ύψους) της φωτιάς που έχει αναπτυχθεί σε κάθε σημείο του μετώπου.
- Τις επικρατούσες συνθήκες ανέμου (ταχύτητα και διεύθυνση) της περιοχής.

2.3. Παράμετροι που χρειάζεται ο προσομοιωτής WSN που αναπτύξαμε για την παραγωγή του θερμοκρασιακού πεδίου.

Οι παράμετροι που χρειάζεται ως είσοδο ο προσομοιωτής μας για να παράγει το θερμοκρασιακό πεδίο γύρω από ένα κινούμενο μέτωπο φωτιάς είναι:

1. Ένα αρχείο το οποίο περιέχει τους χρόνους όπου κάθε σημείο της περιοχής ενδιαφέροντος προσβλήθηκε από τη φωτιά. Τέτοιου είδους αρχεία παράγονται από την εκτέλεση διαφόρων προγραμμάτων εξάπλωσης φωτιάς (Farsite, FSE etc).
2. Ένα αρχείο το οποίο περιέχει τις ταχύτητες των ανέμων που πνέουν σε κάθε σημείο της περιοχής.
3. Ένα αρχείο το οποίο περιέχει τις διευθύνσεις των ανέμων που πνέουν σε κάθε σημείο της περιοχής.
4. Ένα αρχείο το οποίο περιέχει τα μήκη φλόγας που αναπτύσσονται σε κάθε σημείο της περιοχής (εκτιμώμενο από τα προγράμματα εξάπλωσης της φωτιάς).
5. Την θερμοκρασία περιβάλλοντος.
6. Ένα αρχείο το οποίο περιέχει τις συντεταγμένες των αισθητήρων μέσα στο πεδίο.

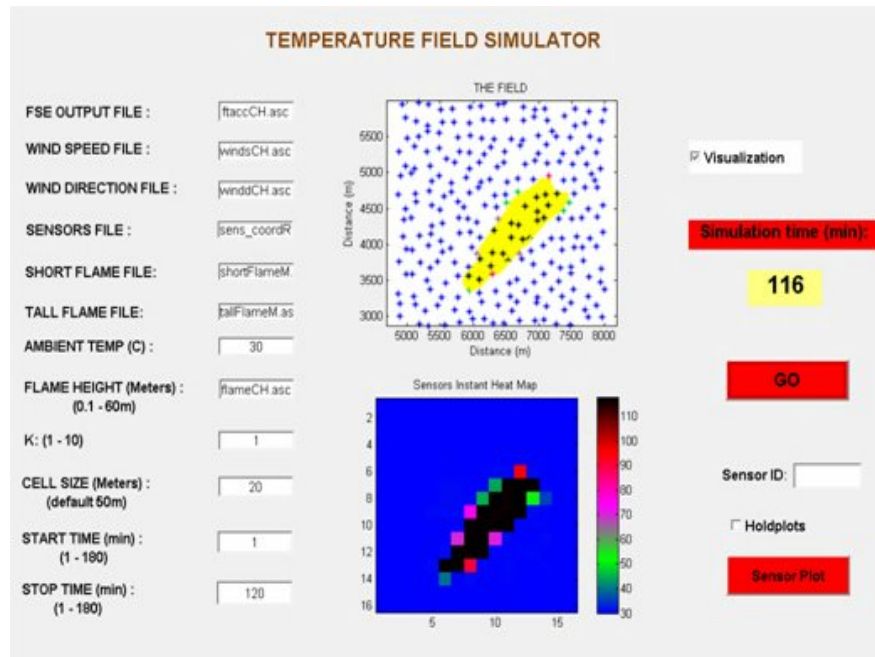
7. Ένα αριθμό για το K , δηλαδή των αριθμό των “κοντινότερων” κανονικών κατανομών οι οποίες θα συμμετέχουν στην εκτίμηση του θερμοκρασιακού πεδίου και επομένως στην θερμοκρασία που θα αντιληφθεί ο κάθε αισθητήρας.
8. Το μέγεθος του κελιού για το οποίο εκτελέστηκε το πρόγραμμα εξάπλωσης.
9. Το χρόνο εκκίνησης της προσομοίωσης.
10. Το χρόνο ολοκλήρωσης της προσομοίωσης (Πόσα λεπτά της εξέλιξης του φαινομένου θα προσομοιωθούν).

Έχοντας αυτό το σύνολο παραμέτρων οι οποίες είναι εύκολο να εκτιμηθούν, μπορούμε να υπολογίσουμε για κάθε χρονική στιγμή το θερμοκρασιακό πεδίο μιας εξελισσόμενης πυρκαγιάς και επομένως τις θερμοκρασίες που θα αντιληφτεί ο κάθε αισθητήρας θερμοκρασίας που είναι τοποθετημένος σε συγκεκριμένα σημεία του πεδίου.

2.4. Γραφικό περιβάλλον και έξοδοι του προσομοιωτή

Με την βοήθεια του προσομοιωτή που αναπτύξαμε, μας παρέχεται η δυνατότητα να οπτικοποιούμε την εξέλιξη του φαινομένου μιας δασικής πυρκαγιάς και να παρατηρούμε για κάθε χρονική στιγμή με την βοήθεια “θερμοκρασιακών εικόνων”, τις θερμοκρασίες που αναμένεται να αισθανθούν οι αισθητήρες οι οποίοι είναι τοποθετημένοι στο πεδίο. Η διαδοχή αυτών των “θερμοκρασιακών εικόνων” που παράγονται κάθε χρονική στιγμή της προσομοίωσης μας παρέχουν καρέ-καρέ ένα είδος “θερμοκρασιακού video”. Με την βοήθεια των γραφικών αποτελεσμάτων, του προσομοιωτή, τόσο κατά την εξέλιξη της προσομοίωσης όσο και μετά την ολοκλήρωσή της, μπορούμε να εξάγουμε άμεσα συμπεράσματα για την ταχύτητα και διεύθυνση εξάπλωσης, καθώς και για το πόσο αποτελεσματική είναι η τοποθέτηση των αισθητήρων μας, για την παρακολούθηση του συγκεκριμένου τύπου πυρκαγιάς. Μετά την ολοκλήρωση κάθε προσομοίωσης, παράγεται ένα αρχείο το οποίο περιέχει το θερμοκρασιακό ιστορικό του κάθε αισθητήρα καθ' όλη τη διάρκεια του φαινομένου. Με βάση αυτό το θερμοκρασιακό ιστορικό, ο προσομοιωτής, μας παρέχει τη δυνατότητα να οπτικοποιήσουμε και να συγκρίνουμε τις καμπύλες θερμοκρασίας – χρόνου για κάθε αισθητήρα από τους αισθητήρες του πεδίου.

Στο σχήμα 3 παρουσιάζεται ένα στιγμιότυπο της εξέλιξης μιας δασικής πυρκαγιάς όπως αυτό εμφανίζεται στο γραφικό περιβάλλον του προσομοιωτή μας. Στους πάνω άξονες παρατηρούμε πως έχει εξελιχθεί η φωτιά μέσα στην περιοχή που είναι εγκατεστημένοι οι ασύρματοι αισθητήριοι κόμβοι θερμοκρασίας. Τα διαφορετικά χρώματα των αισθητήρων (οι μπλε κουκίδες) αντιστοιχούν σε διαφορετικές θερμοκρασίες. Στους κάτω άξονες φαίνεται το πώς το δίκτυο των αισθητήρων που είναι εγκατεστημένο στην περιοχή, αισθάνεται το θερμοκρασιακό πεδίο που παράγεται από την εξελισσόμενη πυρκαγιά. Η χρωματική μπάρα που βρίσκεται δεξιά των κάτω άξονων αντιστοιχίζει τα χρώματα σε θερμοκρασίες Κελσίου ούτως ώστε να μας δίνεται μια οπτική αίσθηση του θερμοκρασιακού πεδίου. Αξίζει να σημειωθεί ότι τα χρώματα των αισθητήρων και στις δύο εικόνες βρίσκονται σε συμφωνία.



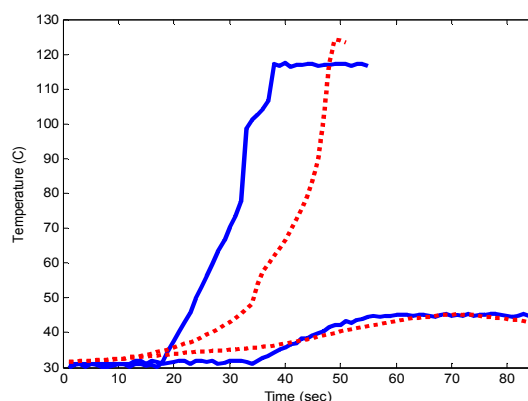
Σχήμα 3: Στιγμιότυπο προσομοίωσης της εξέλιξης δασικής πυρκαγιάς με χρήση του προσομοιωτή που αναπτύξαμε.

2.5. Επαλήθευση των θερμοκρασιακών μοντέλων

Δυστυχώς, δεν υπάρχουν διαθέσιμα δεδομένα από ασύρματα δίκτυα αισθητήρων που παρακολουθούν φαινόμενα δασικών πυρκαγιών. Γι αυτό το λόγο η επιβεβαίωση της σωστής λειτουργίας των θερμοκρασιακών μας μοντέλων είναι πολύ δύσκολο να επιτευχθεί. Παρόλα αυτά προσπαθήσαμε και επαληθεύσαμε τη λειτουργία του μοντέλου μας που εφαρμόζεται για τα μικρά μήκη φλόγας με την βοήθεια μετρήσεων θερμοκρασίας που λήφθηκαν από ασύρματους αισθητήριους κόμβους κατά τη διάρκεια πειραμάτων που έλαβαν χώρα στις εγκαταστάσεις του μεσογειακού αγρονομικού ινστιτούτου των Χανίων (MAICH) της Κρήτης [11].

Παρακάτω παρουσιάζονται τα αποτελέσματα ενός πειράματος που έγινε στο MAICH καθώς και τα αποτελέσματα της προσομοίωσης του συγκεκριμένου πειράματος από τον προσομοιωτή μας. Στο συγκεκριμένο πείραμα χρησιμοποιήθηκαν δύο ασύρματοι κόμβοι S1 και S2 οι οποίοι ήταν εφοδιασμένοι με αισθητήρες θερμοκρασίας και υγρασίας. Ο S1 ήταν τοποθετημένος μέσα στο στρώμα της καύσιμης ύλης (fuel bed) το οποίο περιείχε πευκοβελόνες, ενώ ο S2 ήταν εκτός του στρώματος (25cm από την δεξιά πλευρά του). Η ταχύτητα του ανέμου στο συγκεκριμένο πείραμα ήταν 0.4m/sec και ο το μήκος της φλόγας που αναπτύχθηκε έφτανε τα 60cm.

Στο σχήμα 4, με τις διακεκομμένες γραμμές παρουσιάζονται οι καμπύλες των θερμοκρασιών που κατέγραψαν δύο αισθητήρες S1 και S2 κατά τη διάρκεια του συγκεκριμένου πειράματος. Με τις συνεχείς γραμμές παρουσιάζονται οι θερμοκρασιακές καμπύλες που καταγράφηκαν από τους εικονικούς αισθητήρες κατά την προσομοίωση του συγκεκριμένου πειράματος. Από τα αποτελέσματα της προσομοίωσης παρατηρούμε ότι οι μέγιστες θερμοκρασίες μεταξύ πραγματικών και εικονικών αισθητήρων είναι πολύ κοντά καθώς επίσης και οι κλίσεις τους κατά το ανέβασμα των θερμοκρασιών. Αξίζει να σημειωθεί ότι η εκτιμώμενη θερμοκρασία του αισθητήρα S2 είναι πολύ ρεαλιστική αν σκεφτούμε το γεγονός ότι ο συγκεκριμένος αισθητήρας ήταν έξω από το στρώμα της καύσιμης ύλης και μακριά από την κύρια διεύθυνση της φωτιάς. Αυτό επιβεβαιώνει ότι οι το μοντέλο θερμοκρασίας μας που κάνει χρήση δισδιάστατων κανονικών κατανομών με πλήρη μητρώο συνδιασποράς καθώς και η Mahalanobis απόσταση που χρησιμοποιούμε για την εκτίμηση των θερμοκρασιών δικαιολογείται πλήρως. Τα αποτελέσματα της συγκεκριμένης διπλωματικής θα δημοσιευθούν στο διεθνές συνέδριο EUSIPCO τον Αύγουστο του 2008 [11].



Σχήμα 4: Οι διακεκομμένες (συνεχείς) γραμμές, αναπαριστούν τις καμπύλες θερμοκρασίας συναρτήση του χρόνου των πραγματικών (εικονικών) αισθητήρων. Η καμπύλη του αισθητήρα S1 ανυψώνεται πρώτη.

2.6. Συμπεράσματα

Στα πλαίσια της διπλωματικής αυτής εργασίας αναπτύχθηκε ένα εργαλείο που προσομοιώνει συμπεριφορές ασυρμάτων δικτύων αισθητήρων θερμοκρασίας κατά την εξέλιξη δασικών πυρκαγιών. Η καινοτομία της εργασίας αυτής έγκειται στην προσέγγιση του θερμοκρασιακού πεδίου που παράγεται από ένα εξελισσόμενο μέτωπο πυρκαγιάς με την βοήθεια στατιστικής επεξεργασίας σήματος και ενός συνόλου παραμέτρων που μας παρέχονται από κάποιο πρόγραμμα εξάπλωσης φωτιάς (Farsite, FSE). Ο προσομοιωτής που αναπτύξαμε μπορεί και εκτιμά τις θερμοκρασίες που θα αισθανθεί ο κάθε αισθητήρας που είναι τοποθετημένος στο πεδίο καθώς εξελίσσονται χρονικά και χωρικά τα καταστροφικά φαινόμενα των πυρκαγιών. Με τη

χρήση αυτού του προσομοιωτή μπορούμε να εκτιμήσουμε πώς συγκεκριμένες τοποθετήσεις WSN θερμοκρασίας συμπεριφέρονται σε διαφορετικά σενάρια φωτιάς. Η μελέτη αυτών των συμπεριφορών μπορούν να βοηθήσουν στην ανάπτυξη κατάλληλων αλγορίθμων ανίχνευσης, παρακολούθησης και πρόβλεψης κίνησης τέτοιων φαινομένων καθώς και στην εύρεση βέλτιστων μεθόδων τοποθέτησης συστημάτων WSN για τέτοιου είδους εφαρμογές.

Αναφορές

- [1]. H. Anderson, “Predicting Wind-Driven Wild Land Fire Size and Shape”, For. Serv, Research Paper, INT – 305, February 1983.
- [2]. J. Glasa, L. Halada, “ On elliptical model for forest fire spread modeling and simulation”, Elsevier Science, 6/ 2007.
- [3]. “EUFIRELAB, a wall-less Laboratory for Wildland Fire Sciences and Technologies in the Euro-Mediterranean Region”, Deliverable D-03-06, <http://www.eufirelab.org> (07/02/2008).
- [4]. W. Pitts, E. Braun, R. Peacock, H. Mitler, E. Johnsson, P. Reneke, L. Blevins, “Temperature Uncertainties for Bare-Bead and Aspirated Thermocouple Measurements in Fire Environments,” *In Proc. Annual Conference on Fire Research*, Nov. 1999, pp.15-16.
- [5]. D. Stroup, L. DeLauter, J. Lee, G. Roadarmel “Fire Test of Men’s Suits on Racks”, *Proc. Report of Test FR 4013*, 2001, pp.25.
- [6]. F. Kurt, “Prometheus Fire Growth Model: Design and Incorporation of Spotting and Breaching of Fire Break Functionality,” in Post-Fire Research Workshop April 27, 2005.
- [7]. B. Butler and J. Cohen, “Firefighter Safety Zones: A Theoretical Model Based on Radiative Heating”. *Int. J. Wildland Fire* 8(2):73-77, 1998.
- [8]. H. Anderson, “Calculating Fire Size and Perimeter Growth,” *Journal Fire Management Notes*, 45(3):25-30, 1984.
- [9]. J. Cohen, “Relating flame radiation to home ignition using modeling and experimental crown fires”, *Can. J. For. Res.* 34, 1616 – 1626 p, 2004.
- [10]. M. A. Finney, “FARSITE: Fire Area Simulator – Model Development and Evaluation”, Forest Service, Research Paper, RMRSRP- 4 Revised, March 1998.
- [11]. E. Manolagos, D. Manatakis “Temperature Field Modeling And Simulation Of Wireless Sensor Network Behavior During a Spreading Wildfire,” *in Proc. Annual Conference EUSIPCO*, 25-29 August, 2008.

Ανάπτυξη, υλοποίηση και αξιολόγηση αλγορίθμου δυναμικής ανάθεσης πόρων σε δίκτυα IEEE 802.16e, με χρήση της τεχνικής OFDMA

Διονύσιος Ξενάκης, Δημήτριος Τσόλκας
grad0942@di.uoa.gr, grad0941@di.uoa.gr

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών,
Τμήμα Πληροφορικής και Τηλεπικοινωνιών

Περίληψη : Στην εργασία αυτή προτείνεται μία νέα αρχιτεκτονική επίλυσης και ένα νέο σύνολο μετρικών, για την μοντελοποίηση και την επίλυση του προβλήματος δυναμικής ανάθεσης πόρων (*Dynamic Subchannel Allocation - DSA*), σε δίκτυα τεχνολογίας IEEE 802.16e. Σύμφωνα με την αρχιτεκτονική αυτή, ο αλγόριθμος DSA λειτουργεί ως επέκταση οποιουδήποτε IEEE 802.16 χρονοπρογραμματιστή, έτσι ώστε να εκμεταλλευτεί τα πλεονεκτήματα που προσφέρει η τεχνική πολλαπλής προσπέλασης OFDMA. Από τα αποτελέσματα των προσομοιώσεων προκύπτει ότι η χρήση του προτεινόμενου αλγορίθμου οδηγεί σε σημαντική μείωση των απωλειών λόγω συνθηκών καναλιού.

Λέξεις Κλειδιά : OFDMA, DSA, IEEE 802.16e, απώλειες καναλιού, μεγιστοποίηση ρυθμού.

1. Εισαγωγή - το πρότυπο IEEE 802.16e

Το πρότυπο IEEE 802.16e [1] καθώς και η πρόσφατη τροποποίηση του IEEE 802.16e [2] για υποστήριξη κινητικότητας είναι πολλά υποσχόμενο για την ασύρματη πρόσβαση σε επίπεδο μητροπολιτικών δικτύων (*MAN*), καθώς παρέχει υψηλούς ρυθμούς και εγγυήσεις ποιότητας υπηρεσιών (*QoS*). Το πρότυπο IEEE 802.16e, προβλέπει την χρήση των τεσσάρων ακόλουθων μεθόδων πολλαπλής προσπέλασης σε φυσικό επίπεδο (*PHY*) : single carrier (*SC*), *SCa*, OFDM και OFDMA. Η τεχνική *orthogonal frequency division multiple access (OFDMA)* βασίζεται στην τεχνική OFDM, παρέχοντας όμως ανεξαρτησία ως προς το φαινόμενο της αλληλοπαρεμβολής συμβόλων (*ISI*) και την επιλεκτική εξασθένιση φέρουσας. Η ανεξαρτησία αυτή οφείλεται στην διαμέριση του φάσματος σε *σύνολα από κοινού ορθογώνιων φερουσών*, μικρότερου εύρους ζώνης, τα οποία καλούνται υποκανάλια (*subchannels*)[3]. Όσον αφορά το επίπεδο πρόσβασης μέσου (*MAC*), ο ελάχιστος προς ανάθεση πόρος είναι ένα *slot* [1][2] που ισοδυναμεί με μία χρονοθυρίδα ενός *subchannel*. Η δομή ενός *slot* καθορίζεται από την υιοθετούμενη μέθοδο σχηματισμού των *subchannels*, η οποία δύναται να είναι είτε με κατανεμημένη επιλογή φερουσών από το φάσμα (*DSP*), είτε με επιλογή γειτονικών φερουσών (*ASP*). Η πρώτη μέθοδος παρέχει στους χρήστες ομοιόμορφη κατανομή των παρεμβολών σε κάθε *subchannel*, ενώ η δεύτερη δίνει την ευχέρεια επιλογής *subchannel* με την βέλτιστη ανταπόκριση ως προς την συχνότητα.

Το πρότυπο ενσωματώνει και τις δύο μεθόδους αμφίδρομης επικοινωνίας, *Time Division Duplex (TDD)* και *Frequency Division Duplex (FDD)*, ενώ η διαχείριση και η ανάθεση των πόρων του συστήματος βασίζεται σε πλαίσια (*frames*) που δομούνται από *slots*. Ως προς την αρχιτεκτονική του δικτύου αναφέρουμε ότι η οργάνωση μπορεί να είναι είτε αδόμητη πλεγματική (*mesh*) είτε δομημένη κυψελοειδής *Point to Multipoint (PMP)*.

2. Το πρόβλημα της Δυναμικής Ανάθεσης Πόρων (*DSA*)

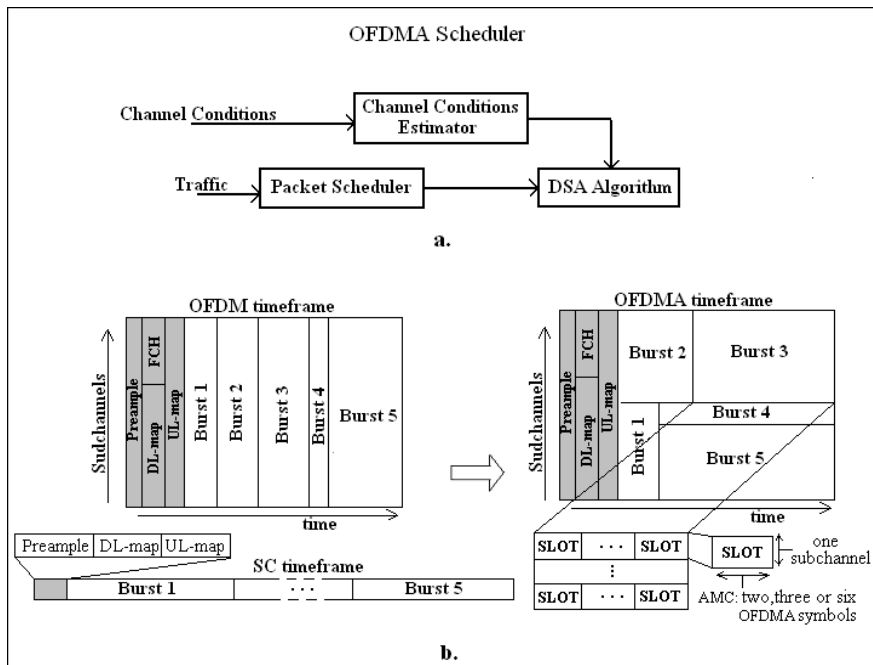
Η ανάλυση που ακολουθεί αφορά *PMP αρχιτεκτονική*, *TDD* αμφίδρομη επικοινωνία, *ASP σχηματισμό subchannels* και *OFDMA PHY* στην καθοδική ζεύξη (*DL*). Ωστόσο, η μοντελοποίηση που ακολουθείται μπορεί να γενικευτεί σε *FDD*, *DSP* και ανοδική ζεύξη (*UL*). Με βάση κάποιον αλγόριθμο, ο σταθμός βάσης (*BS*) αναθέτει μία περιοχή δεδομένων (*data region*) σε κάθε ενεργό κινητό σταθμό (*MS*), λαμβάνοντας υπ' όψιν τις επικρατούσες συνθήκες καναλιού και τις απαιτήσεις του σε εύρος ζώνης. Κάθε *data region*, αποτελείται από ένα σύνολο συνεχόμενων *slots* στο πεδίο του χρόνου και της συχνότητας. Επίσης, κάθε χρήστης δύναται να έχει ένα σύνολο από διαφορετικές συνδέσεις με τον *BS*, οι οποίες πρέπει να λειτουργούν σύμφωνα με την συμφωνημένη ποιότητα υπηρεσίας *QoS*. Η παρούσα εργασία έχει ως στόχο να προτείνει έναν αλγόριθμο δυναμικής ανάθεσης πόρων, που αν και διαδραματίζει σημαντικό ρόλο στην λειτουργία του συστήματος δεν καθορίζεται από το πρότυπο.

Η ανάθεση πόρων μπορεί να γίνεται είτε ανά χρήστη (*Granted per MS – GPMS*) είτε ανά σύνδεση χρήστη (*Granted per Connection - GPC*). Η πρώτη μέθοδος προσφέρει μικρότερη πολυπλοκότητα υλοποίησης λόγω της συνολικής διαχείρισης πόρων κάθε χρήστη, ενώ η δεύτερη παρέχει περισσότερη ευελιξία κατά την διαχείριση των πόρων αν και η πολυπλοκότητα της αυξάνεται δυσανάλογα με το αυξανόμενο πλήθος χρηστών. Στην ανάλυση που ακολουθεί, υιοθετείται η *GPMS* πολιτική διότι α) παρέχει επεκτασιμότητα ως προς το πλήθος των χρηστών, β) με βάση την προτεινόμενη αρχιτεκτονική η ικανοποίηση των αναγκών κάθε σύνδεσης χρήστη λαμβάνεται υπ' όψιν στο στάδιο του χρονοπρογραμματισμού και γ) οι ενεργές συνδέσεις κάθε χρήστη τυγχάνουν της ίδιας ποιότητας καναλιού, επομένως όταν στόχος είναι η επιλογή των καλύτερων *subchannels* η επανάληψη της λήψης απόφασης για το ίδιο πρόβλημα είναι πλεοναστική.

Οι εργασίες [4-7] έχουν επίσης εξετάσει το ανωτέρω πρόβλημα. Στην [4], η ανάθεση πόρων ακολουθεί μία *GPC* πολιτική όπως και στο [5], όπου προτείνεται μία μικτή πολιτική χρονοπρογραμματισμού πακέτων και ανάθεσης *subchannels*, στην οποία για κάθε *subchannel* χρησιμοποιείται μία ουρά προτεραιότητας. Ωστόσο, έχουμε ήδη αναφέρει ότι η πολιτική *GPC* επεκτείνεται δύσκολα ως προς το πλήθος των χρηστών και συνήθως είναι πλεονάζουσα για την επιλογή των καλύτερων

subchannels. Στο [6], ακολουθείται μία προσέγγιση χαμηλής πολυπλοκότητας η οποία κάνει χρήση του μηχανισμού διαχείρισης ισχύος, ενώ στο [7] γίνεται μία cross-layer ανάλυση για συστήματα WiBro με OFDMA PHY.

Στην εργασία αυτή προτείνεται μία νέα προσέγγιση/αρχιτεκτονική επίλυσης του προβλήματος. Σύμφωνα με την αρχιτεκτονική αυτή, η διεργασία ανάθεσης πόρων κατανέμεται μεταξύ των οντοτήτων του χρονοπρογραμματιστή (π.χ. *OFDM*) και του αλγορίθμου *DSA*. Ο χρονοπρογραμματιστής είναι υπεύθυνος για την ικανοποίηση μακροπρόθεσμων αναγκών όπως η ανάθεση του απαραίτητου εύρους ζώνης σε κάθε χρήστη, ο κατάλληλος διαμοιρασμός του πλήθους των πόρων μεταξύ των χρηστών σύμφωνα με την ακολουθούμενη πολιτική, καθώς και η ικανοποίηση των χαρακτηριστικών QoS που απαιτούνται από τις ενεργές συνδέσεις κάθε χρήστη. Τα ανωτέρω χαρακτηριστικά είναι συνήθως γνωστά είτε από την φάση εγκαθίδρυσης σύνδεσης του χρήστη είτε από τις αιτήσεις του MS για εύρος ζώνης. Από την άλλη πλευρά, ο αλγόριθμος *DSA* είναι υπεύθυνος για την λήψη αποφάσεων αναφορικά με την καλύτερη δυνατή ανάθεση *subchannels*, την επιλογή βέλτιστης διαμόρφωσης και σχήματος κωδικοποίησης (χρήση *adaptive modulation and coding scheme*), καθώς και την βέλτιστη διαχείριση των επιπέδων ισχύος (*power control*), λαμβάνοντας υπ' όψιν τις πρόσκαιρες και ταχέως μεταβαλλόμενες συνθήκες καναλιού που επικρατούν για κάθε χρήστη. Οι επικρατούσες συνθήκες καναλιού παρέχονται μέσω μηχανισμού μέτρησης ποιότητας σήματος *SINR* για κάθε *MS*.



Σχήμα 1 : a) Προτεινόμενη Αρχιτεκτονική, b) Αναθέσεις πόρων αρχικού και τελικού frame

Συνολικά, ο αλγόριθμος *DSA* λειτουργεί σε *frame by frame* βάση, μεγιστοποιώντας τον συνολικό ρυθμό του συστήματος, εκμεταλλευόμενος την ευελιξία που παρέχει η τεχνική OFDMA για ανάθεση πόρων στο πεδίο των συχνοτήτων. Για να το επιτύχει, λαμβάνει υπ' όψιν την προηγηθείσα ανάθεση πόρων από τον χρονοπρογραμματιστή και τις τρέχουσες συνθήκες καναλιού ανά χρήστη. Το σχήμα 1a παρουσιάζει την προτεινόμενη αρχιτεκτονική, ενώ το σχήμα 1b παρουσιάζει την δομή και τις αναθέσεις πόρων σε ένα 802.16 frame, όπως αυτές επηρεάζονται από την προτεινόμενη αρχιτεκτονική.

3. Μοντελοποίηση του προβλήματος DSA

Στην ακόλουθη ανάλυση, στόχος μας είναι να επιλύσουμε το πρόβλημα της καλύτερης δυνατής ανάθεσης πόρων. Με βάση τα προηγούμενα, θεωρούμε έναν αλγόριθμο επίλυσης του προβλήματος DSA που ακολουθεί τις δύο ακόλουθες αρχές :

1. Η διαδικασία της ανάθεσης πόρων θα γίνεται βηματικά, όπου σε κάθε βήμα θα οριστικοποιείται η δέσμευση μίας *data region* σε έναν MS.
2. Το πλήθος των *slots* που ανατίθενται στον εν λόγω MS, είναι όσο το δυνατόν πιο κοντά στην αρχική ανάθεση που έγινε από τον χρονοπρογραμματιστή. Επίσης, η διαμόρφωση και η εκπεμπόμενη ισχύς που θα χρησιμοποιηθούν από τον εν λόγω MS, ακολουθούν την αρχική απόφαση του χρονοπρογραμματιστή.

Επομένως, ο αλγόριθμος DSA βασισμένος στις ανωτέρω δύο αρχές, θα πρέπει να αναθέτει τα καλύτερα δυνατά *subchannels* για κάθε χρήστη με στόχο την μεγιστοποίηση του συνολικού ρυθμού του συστήματος για το τρέχον timeframe. Στην υποενότητα 3.1, παραθέτουμε το προτεινόμενο σύνολο ορισμών και μετρικών για την μοντελοποίηση του προβλήματος, ενώ στην υποενότητα 3.2 παρουσιάζουμε έναν εξαντλητικό αλγόριθμο επίλυσης του προβλήματος ο οποίος εξετάζει κάθε δυνατή ανάθεση πόρων και επιλέγει αυτήν που οδηγεί στην μεγιστοποίηση του ρυθμού. Από την παρουσίαση του εξαντλητικού αλγορίθμου, προκύπτει και η διάσταση του προβλήματος από άποψη πολυπλοκότητας. Τέλος, στην υποενότητα 3.3 παρουσιάζουμε κάποιο γενικό πλαίσιο ανάπτυξης αλγορίθμων εκτίμησης της καλύτερης λύσης.

3.1 Ορισμοί και μετρικές

| | | |
|--------|---|---|
| N | : | Πλήθος διαθέσιμων subchannels |
| T | : | Πλήθος χρονοθυρίδων στο timeframe |
| K | : | Πλήθος MSs, που συμμετέχουν στο τρέχον frame |
| MS_i | : | Αναγνωριστικό του i -οστού MS, $1 \leq i \leq K$ |
| d_i | : | Ο αριθμός slots που έχει αναθέσει ο χρονοπρογραμματιστής, στον MS_i |

x_i : Ο αριθμός slots που ανάθεσε τελικά ο αλγόριθμος DSA, στον MS_i
 $SINR(i,j)$: Τιμή του λόγου σήμα προς παρεμβολές, του MS_i στο *subchannel* j

Στην συνέχεια θεωρούμε ότι το απαιτούμενο *Bit Error Rate (BER)* κατά την λήψη των δεδομένων χρησιμοποιώντας κάποιο *subchannel* j , είναι επιτυχής όταν η τιμή $SINR(i,j)$ για τον συγκεκριμένο MS_i , ξεπερνά ένα συγκεκριμένο κατώφλι $SINR$, έστω $threshold(i)$, που εξαρτάται από το σχήμα διαμόρφωσης και κωδικοποίησης (*burst profile*) του MS . Στο πρότυπο, το απαιτούμενο κατώφλι για κάθε *burst profile* δίνεται. Έτσι, όσα *subchannels* έχουν $SINR(i,j) \geq threshold(i)$, θα καλούνται *ενεργά* ενώ όσα όχι, θα καλούνται *ανενεργά*. Το ίδιο θα ισχύει και για τα *slots* των *subchannels* αυτών. Θεωρούμε ότι δεδομένα που αποστέλλονται σε *ενεργά slots* λαμβάνονται σωστά, ενώ όσα αποστέλλονται σε *ανενεργά* όχι.

Διάνυσμα subchannels, δ_i : Διάνυσμα μεγέθους $1 \times N$, που ορίζεται για κάθε MS_i . Κάθε θέση του έχει τιμή 1 (*True*) αν το αντίστοιχο *subchannel* είναι *ενεργό*, αλλιώς έχει τιμή 0 (*False*).

Ρυθμός αποστολής $r_s(i)$: Ανάλογα με το χρησιμοποιούμενο *burst profile*, αντιπροσωπεύει το πλήθος *bits* ανά *slot* που αποστέλλονται.

Παράγων F_i : Ορίζεται ως το ποσοστό των *slots* που ο αλγόριθμος DSA αναθέτει στον MS_i εν συγκρίσει με την αρχική απόφαση του χρονοπρογραμματιστή. Η διαφορά μεταξύ d_i και x_i προκύπτει λόγω του ότι το d_i μπορεί να μην διαιρείται ακριβώς από το πλήθος των *subchannels* που απαρτίζουν την νέα περιοχή που ανατίθεται στον MS_i . Ο παράγων αυτός αντιπροσωπεύει τις πιθανές απώλειες λόγω στρογγυλοποίησης : $F_i = \min\left(\frac{x_i}{d_i}, 1\right)$ (1)

Παράγων συνθηκών καναλιού a_i : Θεωρώντας ότι η περιοχή που αναθέτεται στον MS_i είναι μεταξύ των καναλιών $S(a)$ και $S(b)$, ο παράγων αυτός αποτυπώνει την επίπτωση των συνθηκών καναλιού στον ρυθμό του MS_i . Ισοδυναμεί με το ποσοστό των *ενεργών slots* :

$$a_i = \frac{\sum_{j=S(a)}^{S(b)} \delta_i[j]}{S(b)-S(a)+1} \quad (2)$$

Επιτυχώς ληφθέντα bits B_i : Οι παράγοντες που επηρεάζουν το πλήθος των επιτυχώς ληφθέντων *bits* είναι το πλήθος των εκπεμφθέντων *bits* ($d_i \cdot r_s(i)$), οι επικρατούσες συνθήκες καναλιού (a_i) και η επίπτωση λόγω στρογγυλοποίησης του πλήθους των *slots* με βάση την νέα ανάθεση (F_i). Συνολικά, έχουμε : $B_i = d_i \cdot r_s(i) \cdot a_i \cdot F_i$ (3)
 Το πλήθος των επιτυχώς ληφθέντων *bits* όλων των χρηστών είναι :

$$B_{sys} = \sum_{i=1}^K B_i \quad (4)$$

Συνολικός ρυθμός συστήματος r_{sys} : Αν τ_{frame} αντιπροσωπεύει την χρονική διάρκεια του timeframe, ο συνολικός ρυθμός του συστήματος είναι :

$$r_{sys} = \frac{B_{sys}}{\tau_{frame}} = \frac{\sum_{i=1}^K B_i}{\tau_{frame}} \quad (5)$$

Απώλειες bit λόγω καναλιού (%), L_c :

$$L_c = \frac{\sum_{i=1}^K d_i \cdot r_s(i) - B_{sys}}{\sum_{i=1}^K d_i \cdot r_s(i)} = 1 - \frac{B_{sys}}{\sum_{i=1}^K d_i \cdot r_s(i)} \quad (6)$$

3.2 Εξαντλητικός Αλγόριθμος Επίλυσης

Σύμφωνα με τους παραπάνω ορισμούς και μετρικές, η μεγιστοποίηση του συνολικού ρυθμού του συστήματος ισοδυναμεί με την μεγιστοποίηση του r_{sys} , ή ισοδύναμα με την μεγιστοποίηση του B_{sys} ή ισοδύναμα με την ελαχιστοποίηση των απωλειών bit λόγω καναλιού L_c . Στην συνέχεια, θα γίνει προσπάθεια αποτύπωσης όλων των πιθανών λύσεων του προβλήματος σε κάποια δομή και μέσω ενός εξαντλητικού αλγορίθμου η επιλογή της λύσης που μεγιστοποιεί τον συνολικό ρυθμό του συστήματος.

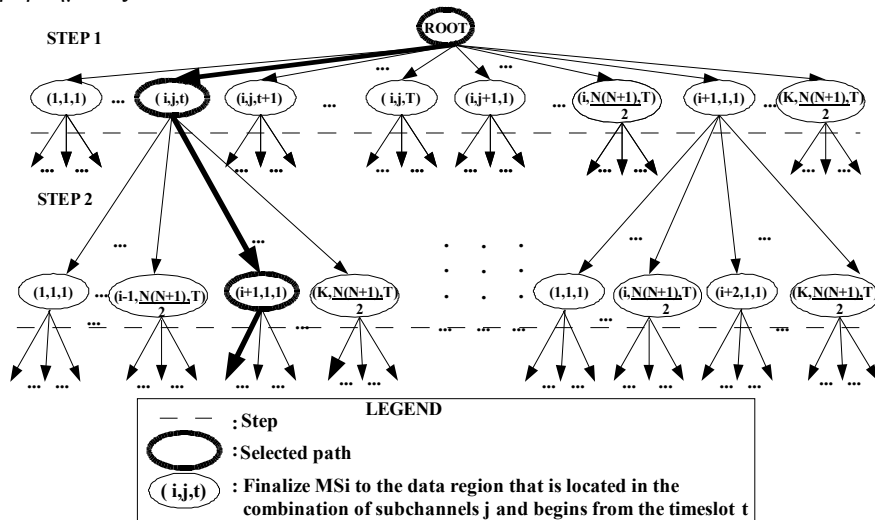
| πλήθος subchannels | αναγνωριστικό συνδυασμού subchannels : [από subchannel , έως subchannel] | | | | | | αριθμός συνδυασμών |
|--------------------|--|---------------|-----|---------------|-----|-------------------|--------------------|
| 1 | 1:[1,1] | (N+1):[2,2] | ... | ... | ... | (N(N+1)/2):[N, N] | N |
| 2 | 2:[1,2] | (N+2):[2,3] | ... | ... | ... | ID:[N-1, N] | N-1 |
| ... | ... | ... | ... | ... | ... | - | ... |
| i | i:[1,i] | (N+i):[2,i+1] | ... | ID:[N-i+1, N] | - | - | N-i+1 |
| ... | ... | ... | ... | - | - | - | ... |
| N-1 | (N-1):[1, N-1] | (2N-1):[2, N] | - | - | - | - | 2 |
| N | N:[1, N] | - | - | - | - | - | 1 |

Σχήμα 2 : Συνδυασμοί και αναγνωριστικά subchannels

Το πρότυπο IEEE 802.16e ορίζει την περιοχή δεδομένων κάθε χρήστη, ως μία συλλογή συνεχόμενων slots στον χρόνο και στην συχνότητα. Επομένως, κάθε πιθανή περιοχή ανάθεσης θα αποτελείται από συνεχόμενα subchannels και συνεχόμενες χρονοθυρίδες, σχηματίζοντας ένα ορθογώνιο. Το σχήμα 2 αποτυπώνει όλους τους πιθανούς συνδυασμούς subchannels, δίνοντας παράλληλα ένα αναγνωριστικό σε καθένα από αυτούς. Ο αριθμός των συνδυασμών αυτών είναι $1 + 2 + \dots + N = \sum_{i=1}^N i = \frac{N(N+1)}{2} = O(N^2)$. Σε καθένα από τους συνδυασμούς αυτούς, δύναται να ορίζονται το πολύ T data regions, που προκύπτουν με βάση το

αρχικό πλήθος *slots* d_i . Συνολικά λοιπόν, ο μέγιστος αριθμός διαθέσιμων προς ανάθεση *data regions* για κάθε MS είναι $T \cdot \frac{N(N+1)}{2} = O(T \cdot N^2)$.

Κάθε ανάθεση πόρων στον MS_i καθορίζεται πλήρως από το αναγνωριστικό του (i), το αναγνωριστικό του συνδυασμού *subchannels* (j) και τον αριθμό της *χρονοθυρίδας* (t) που αρχίζει η *data region* που του ανατέθηκε (το τέλος της οριοθετείται με βάση το d_i). Συμβολίζουμε από εδώ και στο εξής κάθε ανάθεση πόρων με (i,j,t) . Λαμβάνοντας υπ' όψιν ότι στο τέλος κάθε βήματος το πλήθος των εναπομεινάντων MSs μειώνεται κατά ένα, ενώ το πλήθος των διαθέσιμων *data region* μπορεί να παραμείνει το ίδιο (π.χ. γιατί δεν δόθηκαν όλα τα *slots* πάνω στον αντίστοιχο συνδυασμό *subchannels*), έχουμε την δυνατότητα να αποτυπώσουμε όλες τις πιθανές λύσεις του προβλήματος σε μία δενδρική δομή. Η δομή αυτή αποκαλείται *exhaustive DSA tree (E-DSA tree)* και παρουσιάζεται στο Σχήμα 3. Κάθε κόμβος του *E-DSA tree* αναπαριστά μία δέσμευση πόρων (i,j,t) ενώ κάθε μονοπάτι από την ρίζα του δένδρου έως κάποιο φύλλο του, ισοδυναμεί με μία έγκυρη λύση του προβλήματος DSA.



Σχήμα 3 : E-DSA tree

Ένας αλγόριθμος που εξετάζει όλα τα δυνατά μονοπάτια που αποτυπώνονται στο δένδρο αυτό, δύναται να δώσει την λύση που μεγιστοποιεί τον συνολικό ρυθμό του συστήματος. Το πλήθος των μονοπατιών αυτών, καθορίζει και την απαιτούμενη υπολογιστική πολυπλοκότητα του αλγορίθμου αυτού. Έστω ότι το $A(i)$ αναπαριστά το πλήθος των κόμβων του i -οστού επιπέδου του δένδρου, τότε $A(0) = 1$ και $A(1) = K \cdot T \cdot \frac{N(N+1)}{2}$ που ισοδυναμεί με το πλήθος των διαθέσιμων *data regions* επί το πλήθος των MSs. Εφόσον σε κάθε βήμα οι εκκρεμείς MSs μειώνονται κατά ένα, ενώ οι διαθέσιμες περιοχές στην χειρότερη περίπτωση δεν μειώνονται, προκύπτει η αναδρομική σχέση :

$$A(i) = A(i-1) \cdot (K-i+1) \cdot T \cdot \frac{N(N+1)}{2} \quad (7)$$

Συνεπώς, το πλήθος των μονοπατιών του δένδρου $A(K)$, που ισοδυναμεί με την υπολογιστική πολυπλοκότητα του αλγορίθμου, προκύπτει ως ακολούθως :

$$A(K) = \prod_{i=1}^{K-1} A(i) = K! \cdot \left(T \cdot \frac{N(N+1)}{2} \right)^K = O\left(\frac{K! \cdot N^{2K} \cdot T^K}{2^K}\right) \quad (8)$$

3.3 Αλγόριθμοι Εκτίμησης

Λόγω του στενού χρονικού πλαισίου μέσα στο οποίο ο αλγόριθμος πρέπει να δώσει την λύση, η χρήση του εξαντλητικού αλγορίθμου είναι απαγορευτική. Ενώ η χρήση του εξαντλητικού αλγορίθμου δίνει την καλύτερη λύση, μία αποδοτική λύση μπορεί επίσης να εξαχθεί με την χρήση αλγορίθμων εκτίμησης, οι οποίοι αντί της εξέτασης ολόκληρου του μονοπατιού του δέντρου εξετάζουν τους επόμενους h κόμβους κάθε προσβάσιμου μονοπατιού. Αφήνοντας το h ως παράμετρο, παράγεται ένας γενικός αλγόριθμος εκτίμησης *General Estimation Algorithm (GEA)*. Αυτός ο αλγόριθμος συνοψίζεται ως εξής:

1. Θέσε ως αρχικό κόμβο/κόμβο έναρξης, την ρίζα του *E-DSA tree*.
2. Όσο υπάρχουν MSs και πόροι να τους ανατεθούν:
 - 2.1 Υπολόγισε το r_{sys} για τους επόμενους h κόμβους για όλα τα υπό-μονοπάτια που ξεκινούν από τον κόμβο έναρξης.
 - 2.2 Επέλεξε το υπό-μονοπάτι που μεγιστοποιεί το r_{sys} ως το βάθος που εξετάστηκε και προχώρησε στην ανάθεση που αντιπροσωπεύει ο δεύτερος κόμβος του υπό-μονοπατιού.
 - 2.3 Θέσε ως κόμβο έναρξης, τον δεύτερο κόμβο της επιλογής που έγινε στο βήμα 2.2
 - 2.4 Πήγαινε στο βήμα 2.

Στο $(K-h+1)$ βήμα, η διαδικασία εκτίμησης τελειώνει καθώς συναντάται το τελευταίο επίπεδο του δέντρου. Ένας τέτοιος αλγόριθμος προσφέρει δυνατότητα *trade-off* μεταξύ υπολογιστικής πολυπλοκότητας και ακρίβειας εκτίμησης της λύσης. Στην ενότητα 5, θα καταστεί σαφές ότι αυτή η ανταλλαγή δεν κοστίζει όσο εκ πρώτης όψεως φαίνεται. Στην συνέχεια υπολογίζουμε την πολυπλοκότητα των αλγορίθμων *GEA* με παράμετρο h .

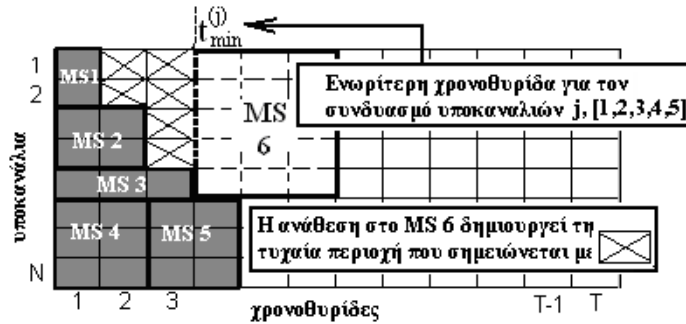
Σε κάθε βήμα, ο *GEA* εξετάζει υπό-δέντρα βάθους h , του *E-DSA tree*. Ως εκ τούτου, στο πρώτο βήμα αυτός ο αλγόριθμος εξετάζει τις $A(h)$ υπό-μονοπάτια, στο δεύτερο βήμα εξετάζει $A(h+1)/A(1)$ υπό-μονοπάτια και γενικά, στο βήμα j εξετάζει $A(h+j)/A(j)$ υπό-μονοπάτια. Κάθε όρος του ακόλουθου αθροίσματος αντιπροσωπεύει τον αριθμό των υπό-μονοπατιών που εξετάζονται σε κάθε βήμα, ενώ το ίδιο το άθροισμα δίνει τα υπό-μονοπάτια που εξετάζονται συνολικά:

$$\sum_{j=0}^{K-h} \frac{A(h+j)}{A(j)} = \sum_{j=0}^{K-h} \frac{(h+j)! \cdot (T \cdot N \cdot (N+1)/2)^{h+j}}{j! \cdot (T \cdot N \cdot (N+1)/2)^j} = (T \cdot N \cdot (N+1)/2)^k \cdot \sum_{j=0}^{K-h} \frac{(h+j)!}{j!} = O\left(\frac{T^k \cdot N^{2k}}{2^k} \cdot \frac{(K-h+1) \cdot K!}{(K-h)!}\right) \quad (9)$$

Η παράμετρος h ρυθμίζει την τελική πολυπλοκότητα, ενώ για $h=K$ ο GEA συμπίπτει με τον εξαντλητικό αλγόριθμο.

4 Προτεινόμενος αλγόριθμος επίλυσης προβλήματος

Στη συνέχεια εξετάζεται η απόδοση του αλγορίθμου GEA με την μικρότερη υπολογιστική πολυπλοκότητα ($h=1$). Σε κάθε βήμα του αλγορίθμου αυτού, οι αναθέσεις που εξετάζονται είναι μόνο αυτές που ανήκουν στο αντίστοιχο επίπεδο του δέντρου. Έτσι, σε κάθε βήμα η επιλογή του υπό-μονοπατιού που μεγιστοποιεί τον όρο r_{sys} , ισοδυναμεί στην επιλογή της ανάθεσης που οδηγεί στο μεγαλύτερο B_i . Ο αλγόριθμος αυτός είναι *άπληστος (greedy)* και δεν προβλέπει τις επιπτώσεις, κάποιας ανάθεσης στις επόμενες. Η επίδραση της απληστίας του όμως μπορεί να ελαχιστοποιηθεί, επιλέγοντας σε κάθε βήμα ένα κατάλληλο προς εξέταση υποσύνολο αναθέσεων, όπως αναλύεται παρακάτω.



Σχήμα 4 : Θέματα αναθέσεων data regions

Όπως σημειώθηκε στην ενότητα 3, υπάρχουν περισσότερες από μία *data regions* προς ανάθεση, που εντοπίζονται στον ίδιο συνδυασμό *subchannels j* και διαφέρουν ως προς την εναρκτήρια χρονοθυρίδα τους. Οι περιοχές αυτές απολαμβάνουν τις ίδιες συνθήκες καναλιού και επομένως, για $h=1$ η εξέταση των αναθέσεων αυτών οδηγεί στο ίδιο B_i (πλεονασμός). Ως εκ τούτου, για κάθε συνδυασμό *subchannels j* θα εξετάζεται μόνο η ανάθεση $(i,j,t_{min}^{(j)})$, όπου το $t_{min}^{(j)}$ αντιπροσωπεύει την ενωρίτερη διαθέσιμη χρονοθυρίδα για τον j -οστό συνδυασμό *subchannels* (Σχήμα 4). Επίσης, μερικές από τις $(i,j,t_{min}^{(j)})$ αναθέσεις μπορεί να οδηγήσουν σε δημιουργία μη αναθετημένων περιοχών τυχαίου σχήματος (Σχήμα 4). Υπάρχουν δύο τρόποι διαχείρισης των περιοχών αυτών, είτε χαρτογράφηση τους για επόμενη ανάθεση, είτε μη ανάθεση τους. Για την αποφυγή του φαινομένου, επιτρέπουμε έλεγχο μόνο των $(i,j,t_{min}^{(j)})$ αναθέσεων, όπου $t_{min}^{(j)}$ αντιπροσωπεύει την ενωρίτερη διαθέσιμη χρονοθυρίδα οποιουδήποτε *subchannel*. Σαν αποτέλεσμα αυτής της τεχνικής έχουμε τη μείωση του αριθμού των αναθέσεων που ελέγχονται και αναθέσεις χωρίς όπισθεν κενά.

Συνολικά ο προτεινόμενος αλγόριθμος σε κάθε βήμα εξετάζει όλες τις έγκυρες (i,j,t_{min}) αναθέσεις και προχωρά στην ανάθεση που μεγιστοποιεί το B_i . Αν υπάρχουν δύο ή περισσότερες αναθέσεις που δίνουν το ίδιο B_i τότε ένα πλήθος κριτηρίων επιλογής χρησιμοποιούνται. Τα κριτήρια αυτά μπορεί να είναι : (i) η ελαχιστοποίηση της διαφοράς αρχικών και αναθετημένων $slot |x_i - d_i|$, (ii) το πλήθος των ενεργών slots των υπολοίπων χρηστών, στην συγκεκριμένη περιοχή, κ.α. Στην εργασία δίνεται ο ένας αποδοτικός τρόπος μέτρησης της σημαντικότητας κάθε slot για κάθε *subchannel*, που βασίζεται στην αρχική ανάθεση *slots ανά subchannel και ανά MS*, στις συνθήκες καναλιού κάθε *MS* και στις αναθέσεις περιοχών που προκύπτουν σε κάθε βήμα του DSA αλγορίθμου. Βάση της παραπάνω ανάλυσης, ο προτεινόμενος αλγόριθμος συνοψίζεται ως εξής:

1. Όσο υπάρχουν MSs και πόροι να τους ανατεθούν:

1.1 Για κάθε εναπομείναντα MS_i

1.1.1 Για κάθε συνδυασμό από υποκανάλια j

1.1.2 Υπολόγισε το εκτιμώμενο B_i για όλες τις (i,j,t_{min}) αναθέσεις

1.2 Προχώρησε στην ανάθεση που i) μεγιστοποιεί το B_i , ii) ελαχιστοποιεί το $|x_i - d_i|$, iii) χρησιμοποιεί τα λιγότερο σημαντικά slots για τους εναπομείναντες MSs.

Ως περίπτωση του αλγορίθμου $GEA(h=1)$, η πολυπλοκότητα του προτεινόμενου αλγορίθμου θα ήταν $O(T \cdot K^2 \cdot N^2)$. Λαμβάνοντας υπ' όψιν όμως πως μόνο οι (i,j,t_{min}) αναθέσεις εξετάζονται σε κάθε βήμα, η υπολογιστική πολυπλοκότητα του αλγορίθμου έχει ως άνω φράγμα το $O(K^2 \cdot N^2)$ και ως κάτω φράγμα το $O(K^2 \cdot N)$. Σε επόμενη δημοσίευση θα δειχθεί ότι η πολυπλοκότητα του αλγορίθμου δύναται να μειωθεί ακόμα περισσότερο εξετάζοντας τις αναθέσεις με δομημένο τρόπο και αποφεύγοντας πλεοναστικούς ελέγχους.

5. Μοντέλο προσομοίωσης – Αποτελέσματα

Για την αποτίμηση της βελτίωσης που προσφέρει η χρήση του προτεινόμενου αλγορίθμου εν συγκρίσει με την επίδοση ενός χρονοπρογραμματιστή που δεν έχει την δυνατότητα αυτή, δημιουργήσαμε ένα μοντέλο προσομοίωσης σε C++. Τα κύρια χαρακτηριστικά του μοντέλου αυτού συνοψίζονται ως εξής :

Γεννήτορας Κίνησης : Προσομοιώνει την πραγματική κίνηση φορτίου που έχει κάποιος MS που χρησιμοποιεί μία UGS σύνδεση σταθερού ρυθμού 64Kbps και μέγιστη αποδεκτή καθυστέρηση 20msec, μία rtPS σύνδεση με μέσο ρυθμό 256Kbps 64Kbps και μέγιστη αποδεκτή καθυστέρηση 40msec, μία nrtPS σύνδεση με μέσο ρυθμό 128Kbps και μία BE σύνδεση με μέσο ρυθμό 128Kbps.

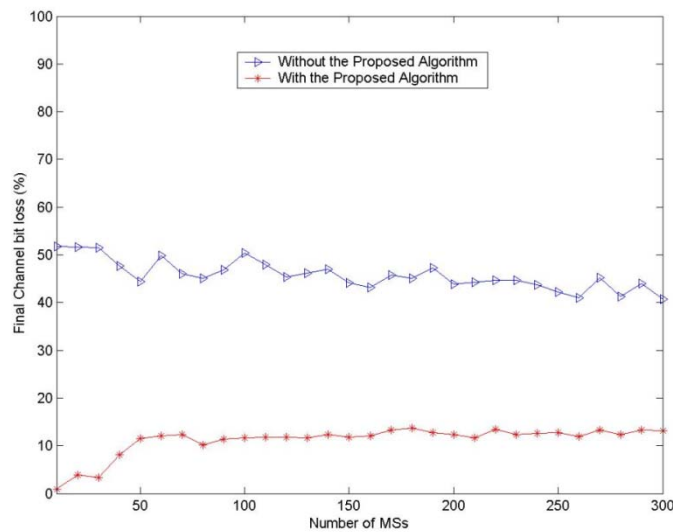
Χρονοπρογραμματιστής : Σύμφωνα με την κίνηση που παράγεται από τον γεννήτορα, τις απαιτήσεις κάθε σύνδεσης χρήστη σε QoS και bandwidth, παράγει ένα πλαίσιο

κίνησης. Οι διαθέσιμες διαμορφώσεις είναι *QPSK*, *16-QAM* και *64-QAM* ενώ ο χρονοπρογραμματιστής που χρησιμοποιείται είναι ο FRTS [8].

Μοντέλο Καναλιού : Παρέχει ένα SINR δiάνυσμα ανά MS, με τιμές για κάθε subchannel. Οι τιμές SINR προκύπτουν από μία μακροβιανή πηγή διακριτού χρόνου στην οποία γειτονικές καταστάσεις διαφέρουν κατά 0.25 dB και παίρνουν τιμές μεταξύ 8.5 και 25 dB.

Για τον υπολογισμό της επίδοσης του αλγορίθμου, υλοποιήθηκαν δύο σενάρια προσομοίωσης. Το πρώτο εξετάζει την επίπτωση του αυξανόμενου πλήθους χρηστών στην επίδοση του αλγορίθμου θεωρώντας σταθερές μέσες αρχικές απώλειες καναλιού ίσες με 45%. Το δεύτερο εξετάζει την επίδοση του αλγορίθμου για σταθερό πλήθος χρηστών (150) καθώς οι μέσες αρχικές απώλειες καναλιού L_c^{init} αυξάνονται.

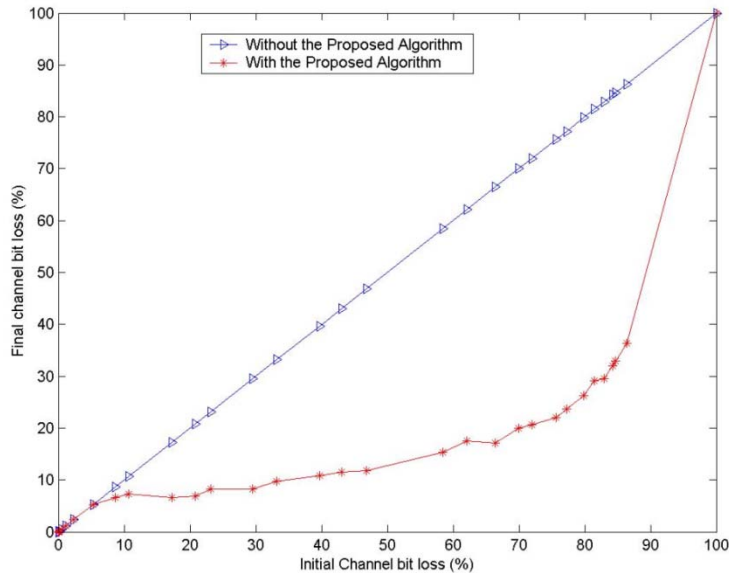
Τα αποτελέσματα του πρώτου σεναρίου παρουσιάζονται στο Σχήμα 5. Προκύπτει ότι από την στιγμή που το frame γεμίζει (περίπου 40 χρήστες), η επίδοση του αλγορίθμου είναι ανεξάρτητη του πλήθους των χρηστών. Μια συνεχής βελτίωση στις απώλειες bit λόγω συνθηκών καναλιού της τάξης τους 35% επιτυγχάνεται.



Σχήμα 5 : Τελικές απώλειες καναλιού για αυξανόμενο πλήθος χρηστών

Τα αποτελέσματα του δεύτερου σεναρίου αποτυπώνονται στο σχήμα 6. Παρατηρούμε ότι για μέτριες προς κακές συνθήκες καναλιού ($L_c^{init} < 75\%$), ενώ οι απώλειες καναλιού χωρίς την χρήση του προτεινόμενου αλγορίθμου αυξάνονται, ο προτεινόμενος αλγόριθμος διατηρεί τις τελικές απώλειες bit λόγω συνθηκών καναλιού σε αρκετά χαμηλά επίπεδα της τάξης του 10 – 20%, παρέχοντας βελτίωση

ως και 55%. Για πολύ κακές συνθήκες καναλιού ($L_c^{init} > 75\%$) η επίδοση του αλγορίθμου είναι επίσης υψηλή.



Σχήμα 6 : Τελικές απώλειες καναλιού για αυξανόμενες αρχικές απώλειες καναλιού

Από τα ανωτέρω προκύπτει ότι ο προτεινόμενος αλγόριθμος εκμεταλλεύεται σε ικανοποιητικό βαθμό τα πλεονεκτήματα που παρέχει η μέθοδος OFDMA, διατηρώντας τις απώλειες σε συγκριτικά χαμηλές τιμές. Λόγω των εξαιρετικά υψηλών επιδόσεων που παρουσιάζει, οδηγεί το σύστημα σε ευρωστία και μπορεί ακόμα να εξαλείψει τις επιπτώσεις κάποιας λάθος επιλογής διαμόρφωσης.

6. Συμπεράσματα

Στην παρούσα εργασία προτάθηκε ένα νέο σύνολο παραμέτρων και μία νέα αρχιτεκτονική/προσέγγιση του προβλήματος DSA. Μέσω της χαρτογράφησης όλων των πιθανών δεσμεύσεων πόρων στο E-DSA tree, έγινε δυνατή η εύρεση της λύσης που μεγιστοποιεί τον ρυθμό του συστήματος με την χρήση ενός εξαντλητικού αλγορίθμου. Ωστόσο, η πολυπλοκότητα του αλγορίθμου αυτού είναι απαγορευτική με αποτέλεσμα η χρήση προσεγγιστικών αλγορίθμων να είναι προτιμότερη καθώς παρέχουν ανταλλαγή πολυπλοκότητας και ακρίβειας πρόβλεψης της λύσης είτε μέσω αποφυγής πλεοναστικών ελέγχων είτε μέσω εισαγωγής περιορισμών.

Ο προτεινόμενος αλγόριθμος ανήκει στην κατηγορία των αλγορίθμων εκτίμησης, παρουσιάζει ιδιαίτερες υψηλές επιδόσεις, επεκτείνει υπάρχοντες χρονοπρογραμματιστές υψηλών επιδόσεων, δύναται να ενσωματώσει μηχανισμούς

adaptive modulation and coding scheme και *power control* και έχει περιθώρια βελτίωσης της απαιτούμενης υπολογιστική του πολυπλοκότητας καθότι βασίζεται σε μαθηματική μοντελοποίηση.

Η παρούσα εργασία έχει δημοσιευθεί στο *IEEE International Symposium on Wireless Pervasive Computing (ISWPC) 2008*, Santorini, Greece, May 2008.

Αναφορές

- [1] IEEE Std 802.16, “IEEE Standard for Local and metropolitan area networks. Part 16: Air Interface for Fixed Broadband Wireless Access Systems”, October 2004.
- [2] IEEE Std 802.16e, “IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems. Amendment 2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands and Corrigendum 1”, February 2006.
- [3] Syed Hussain Ali, Ki-Dong Lee, and Victor C.M. Leung, “Dynamic Resource Allocation in OFDMA Wireless Metropolitan Area Networks” *Wireless communications, IEEE*, vol. 14 Issue 1, p 6-13, Feb. 2007.
- [4] M. Hawa, D.W. Petr. “Quality of Service Scheduling in Cable and Broadband Wireless Access Systems” 10th IEEE International workshop on Quality of Service, May 2002, pp 247-255.
- [5] Lihua Wan, Wenchao Ma, Zihua Guo, “A Cross-layer Packet Scheduling and Subchannel Allocation Scheme in 802.16e OFDMA System” *Wireless Communications and Networking Conference, WCNC 2007 IEEE*, p 1865-1870, March 2007.
- [6] Ian C. Wong, Zukang Shen, Brian L. Evans, and Jeffrey G. Andrews, “A Low Complexity Algorithm for Proportional Resource Allocation in OFDMA Systems” *Signal Processing Systems, SIPS 2004, IEEE Workshop 2004*, p 1- 6, Oct. 2004.
- [7] Taesoo Kwon, Howon Lee, Sik Choi, Juyeop Kim, and Dong-Ho Cho, “Design and Implementation of a Simulator Based on a Cross-Layer Protocol between MAC and PHY Layers in a WiBro Compatible IEEE 802.16e OFDMA System” *Communications Magazine, IEEE*, vol. 43 issue 12, p 136-146, December 2005.
- [8] S. Xergias, N. Passas, and L. Merakos, “Flexible Resource Allocation in IEEE 802.16 Wireless Metropolitan Area Networks”, *Proc. 14th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN)*, Chania, Greece, September 2005.

Βελτιστοποίηση ερωτημάτων σε κατανεμημένες βάσεις

Εύα Σιταρίδη

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών
Τμήμα Πληροφορικής και Τηλεπικοινωνιών
Πανεπιστημιούπολη, Ιλίσια, 15784, Αθήνα

Περίληψη

Οι αυξημένες ανάγκες στη διαθεσιμότητα και τον όγκο των δεδομένων έχει ως αποτέλεσμα το σχηματισμό περίπλοκων δικτύων βάσεων δεδομένων όπου τα δεδομένα κατανέμονται μεταξύ τους. Η κύρια πηγή προβλημάτων σε αυτόν τον τομέα είναι η αυτονομία των κόμβων, που έχει ως άμεση συνέπεια την έλλειψη πληροφορίας για τα χαρακτηριστικά τους, όπως για παράδειγμα την υπολογιστική ικανότητά τους, το κόστος, την ποιότητα των αποτελεσμάτων που παράγουν και το διαθέσιμο εύρος ζώνης τους. Σε αυτή την εργασία, έχει υλοποιηθεί ένας αλγόριθμος εμπνευσμένος από την τεχνολογία ηλεκτρονικού εμπορίου, ο οποίος θεωρεί τα ερωτήματα εμπορικά αγαθά και τη βελτιστοποίηση ερωτημάτων εμπορική διαπραγμάτευση.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Βάσεις δεδομένων

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Κατανεμημένες βάσεις, βελτιστοποίηση ερωτημάτων

1 Εισαγωγή

Οι σύγχρονες αυξημένες απαιτήσεις σε ό,τι αφορά τον όγκο και την διαθεσιμότητα των πληροφοριών έχουν οδηγήσει στον σχηματισμό περίπλοκων δικτύων βάσεων δεδομένων όπου τα δεδομένα κατανέμονται μεταξύ εκατοντάδων κόμβων. Σε τέτοια δίκτυα, ο υπολογισμός του αποτελέσματος ενός ερωτήματος απαιτεί αρχικά τη διάσπασή του σε μικρότερα τμήματα-υποερωτήματα, την ανάθεση και εκτέλεση των τμημάτων αυτών σε διάφορους απομακρυσμένους κόμβους και τέλος τον υπολογισμό του αποτελέσματος του αρχικού ερωτήματος με κατάλληλη συγχώνευση των επιμέρους αποτελεσμάτων των υποτμημάτων. Η έκταση των σύγχρονων κατανεμημένων συστημάτων σε συνδυασμό με την επιθυμητή πλήρη αυτονομία του κάθε κόμβου σε ένα κατανεμημένο σύστημα αλλά και την πιθανή ετερογένεια των επιμέρους συστημάτων, θέτουν σημαντικές προκλήσεις στα θέματα επεξεργασίας και βελτιστοποίησης ερωτημάτων. Η κύρια πηγή προβλημάτων είναι η αυτονομία των κόμβων, που έχει ως άμεση συνέπεια την μερική ή και ολική έλλειψη έγκυρης και επίκαιρης πληροφορίας για συγκεκριμένους απομακρυσμένους κόμβους σε ότι αφορά την ποιότητα, την διαθεσιμότητα, την ποσότητα και τα χαρακτηριστικά των

⁰ Γιάννης Ιωαννίδης yannis@di.uoa.gr
Φραγκίσκος Πεντάρης frank@di.uoa.gr

δεδομένων και πόρων που διαθέτουν. Για παράδειγμα, σε ένα δίκτυο αυτόνομων ΣΔΒΔ, πληροφορίες όπως η τρέχουσα διαθέσιμη υπολογιστική ικανότητά των κόμβων ή ο χρόνος της τελευταίας ανανέωσης των δεδομένων τους συνήθως δεν είναι διαθέσιμα σε τρίτους κόμβους.

Η προηγούμενη δουλειά πάνω στο θέμα έχει δείξει ότι οι παραδοσιακές τεχνικές βελτιστοποίησης ερωτημάτων δεν λειτουργούν καλά στις συνεργασίες αυτόνομων συστημάτων βάσεων δεδομένων. Οι μοναδικοί κατάλληλοι αλγόριθμοι οι οποίοι έχουν προταθεί είναι ο Magirosa και μια συγκεκριμένη παραλλαγή του επαναληπτικού δυναμικού προγραμματισμού.

Στην παρούσα εργασία, έχει υλοποιηθεί τμήμα ενός αλγορίθμου που προτείνει μια νέα προσέγγιση εμπνευσμένη από την τεχνολογία ηλεκτρονικού εμπορίου που ανταποκρίνεται στις ανάγκες καταναμημένων βάσεων μεγάλης κλίμακας και σέβεται απόλυτα την αυτονομία τους. Σύμφωνα με αυτόν τα ερωτήματα αντιμετωπίζονται ως εμπορικά αγαθά και η βελτιστοποίηση ερωτημάτων ως εμπορική διαπραγμάτευση. Οι απαντήσεις σε τμήματα του αρχικού ερωτήματος και οι διαδικασίες συγχώνευσης αυτών των τμημάτων ανταλλάσσονται μεταξύ των κόμβων μέχρι να επιτευχθεί συμφωνία από τους κόμβους που συμμετέχουν στη διαδικασία βελτιστοποίησης.

Σε ό,τι αφορά την υλοποίηση του ανωτέρου αλγορίθμου, το οποίο ήταν και ο κύριος στόχος της πτυχιακής μας εργασίας, υλοποιήθηκε ο εξυπηρετητής καταλόγου, ο οποίος αποτελεί τον χρυσό οδηγό του δικτύου των ΣΔΒΔ καθώς περιέχει πληροφορίες για το ποιοί κόμβοι περιέχουν ποιες σχέσεις. Οι κόμβοι χρησιμοποιούν τον κατάλογο αυτό ώστε να γνωρίζουν ποιοι μπορεί να ενδιαφέρονται να κάνουν προσφορές για ένα τμήμα ενός ερωτήματος. Επίσης υλοποιήθηκε ο αλγόριθμος που εκτελείται σε κάθε κόμβο-πωλητή και αποτελεί παραλλαγή του δυναμικού προγραμματισμού. Ένα μεγάλο τμήμα της υπόλοιπης υλοποίησης έγινε από τους συναδέλφους Σπύρο Μπούτση και Παναγιώτα Κολτσίδα που ολοκλήρωσαν τις σπουδές τους νωρίτερα. Τέλος, πραγματοποιήθηκε πειραματική αξιολόγηση και σύγκριση της γενικής απόδοσης του αλγορίθμου σε σχέση με τους άλλους γνωστούς της περιοχής και εξάχθηκαν αποτελέσματα τα οποία κρίνουμε ως πολύ ενδιαφέροντα.

2 Ο αλγόριθμος βελτιστοποίησης ερωτημάτων

Ο υλοποιούμενος αλγόριθμος βελτιστοποίησης είναι στην πραγματικότητα ένα είδος παζαριού μεταξύ του κόμβου-αγοραστή και των κόμβων-πωλητών. Ο κόμβος-αγοραστής κάνει αίτηση για βελτιστοποίηση που αφορά συγκεκριμένα ερωτήματα και οι κόμβοι-πωλητές κάνουν αντιπροσφορές διαφορετικού κόστους για ορισμένα υποερωτήματα. Η διαφορά μεταξύ της προσέγγισης αυτού του αλγορίθμου με αυτή ενός γενικού πλαισίου εμπορικών συναλλαγών, είναι ότι σε κάθε επανάληψη αυτής της διαπραγμάτευσης τα ερωτήματα που συμμετέχουν στη διαπραγμάτευση αλλάζουν, δεδομένου ότι ο κόμβος-αγοραστής και οι κόμβοι-πωλητές μπορεί να προσδιορίσουν σε κάθε βήμα του αλγορίθμου νέα ενδιαφέροντα ερωτήματα για συμμετοχή στη διαδικασία βελτιστοποίησης/διαπραγμάτευσης. Αυτή η διαφορά έχει ως άμεση συνέπεια να αλλάζουν οι κόμβοι-πωλητές σε κάθε βήμα της διαπραγμάτευσης, δεδομένου ότι τα νεοεισερχόμενα στη διαδικασία ερωτήματα μπορούν να προσ-

φερθούν καλύτερα με χαμηλότερο κόστος από άλλους κόμβους του κατανεμημένου δικτύου.

Οι κόμβοι-αγοραστές είναι αυτοί που είτε δεν επιθυμούν ή απλά δεν μπορούν αποδοτικά να εκτελέσουν τοπικά ένα ερώτημα λόγω έλλειψης πόρων όπως είναι για παράδειγμα τα δεδομένα, η μνήμη RAM και οι κύκλοι CPU. Οι κόμβοι-πωλητές κάνουν προσφορές στους κόμβους-αγοραστές για τμήματα του ερωτήματος για το οποίο πραγματοποιείται η βελτιστοποίηση. Ανάλογα με τα δεδομένα που αποθηκεύει ένας κόμβος και το ερώτημα που βελτιστοποιείται μπορεί να παίζει το ρόλο του αγοραστή ή του πωλητή. Υλοποιείται λοιπόν ένα νέο συγκεκριμένο πλαίσιο που έχει ως βασικές συνιστώσες τους αλγόριθμους βελτιστοποίησης που εκτελούνται στους κόμβους-αγοραστές και πωλητές, τη μετρική κόστους για τα ερωτήματα και τα περιεχόμενα των ανταλλασσόμενων μηνυμάτων διαπραγμάτευσης. Το αποτέλεσμα είναι ένας μηχανισμός εμπορικών συναλλαγών ερωτημάτων, καθώς και τμημάτων αυτών. Για κάθε αρχικό ερώτημα, οι κόμβοι διαπραγματεύονται και ανταλλάσσουν τμήματά του προκειμένου να βρουν το καλύτερο δυνατό πλάνο εκτέλεσης του ερωτήματος αυτού.

Ο παρακάτω πίνακας παρουσιάζει λεπτομερώς τα βήματα του κατανεμημένου αλγορίθμου βελτιστοποίησης για τους κυρίως συμμετέχοντες κόμβους. Ο εξυπρέτης καταλόγου συμμετέχει κάθε φορά που ένας κόμβος θέλει να δει ποιοι μπορούν να του θέσουν προσφορές για κάποιο από τα ενδιφέροντα ερωτήματα.

| Αλγόριθμος κόμβου-αγοραστή | Αλγόριθμος κόμβου-πωλητή |
|---|---|
| <p>A0. Αρχικοποίηση, $Q = \{q, C\}$</p> <p>A1. Εκτίμηση των τιμών των ερωτημάτων στο σύνολο Q, χρησιμοποιώντας στρατηγική εμπορικών συναλλαγών</p> <p>A2. Αίτηση για προσφορές για τα ερωτήματα στο σύνολο Q</p> <p>A3. Επιλογή των καλύτερων προσφορών q_i,εί χρησιμοποιώντας μια από τις τρεις μεθόδους (προσφορά, δημοπρασία, διαπραγμάτευση) του πλαισίου εμπορικών συναλλαγών ερωτημάτων</p> <p>A4. Εύρεση των πιθανών πλάνων εκτέλεσης P_m και το εκτιμώμενο κόστος τους C_m χρησιμοποιώντας τις καλύτερες προσφορές</p> | <p>Π1. Για κάθε ερώτημα q στο σύνολο Q:</p> <p>Π2.1. Εύρεση υποερωτημάτων q_k του q που μπορούν να απαντηθούν τοπικά</p> <p>Π2.2. Εκτίμηση κόστους c_k των q_k</p> <p>Π2.3. Εύρεση επιπλέον ερωτημάτων πιθανά χρήσιμα στον κόμβο-αγοραστή.</p> <p>Π3. Υποβολή προσφορών για ορισμένα από τα υποερωτήματα των βημάτων Π2.2 και Π2.3</p> |

| Αλγόριθμος κόμβουαγοραστή | Αλγόριθμος κόμβου-πωλητή |
|---|--------------------------|
| <p>A5. Εύρεση πιθανών υποερωτημάτων q_e και το εκτιμώμενο κόστος τους c_e που θα μπορούσαν, αν είναι διαθέσιμα, να χρησιμοποιηθούν στο βήμα A4.</p> <p>A6. Ενημέρωση του συνόλου Q με την εισαγωγή των ζευγών (q_e, c_e).</p> <p>A7. Έστω το P^* το καλύτερο από τα πλάνα εκτέλεσης P_m. Εάν το P^* είναι καλύτερο από αυτό της προηγούμενης επανάληψης του αλγορίθμου, ή εάν το βήμα B6 τροποποίησε το καθορισμένο χ, ο αλγόριθμος επιστρέφει στο βήμα B1.</p> <p>A8. Ενημερώστε των κόμβων-πωλητών, ποια ερωτήματα χρησιμοποιούνται στο καλύτερο πλάνο εκτέλεσης P^*, για να εκτελέσουν αυτά τα υποερωτημάτα.</p> | |

3 Υλοποίηση του αλγορίθμου

Το κύριο τμήμα της εργασίας αφορούσε την υλοποίηση του εξυπηρετή καταλόγου, ο οποίος αποτελεί τον χρυσό οδηγό του δικτύου των ΣΔΒΔ που περιέχει πληροφορίες για το ποιοι κόμβοι περιέχουν ποιές σχέσεις. Οι κόμβοι απευθύνονται σε αυτόν για να γνωρίζουν ποιοι μπορεί να ενδιαφέρονται να κάνουν προσφορές για ένα τμήμα ενός ερωτήματος. Επίσης υλοποιήθηκε ο αλγόριθμος που εκτελείται σε κάθε κόμβο-πωλητή και αποτελεί παραλλαγή του δυναμικού προγραμματισμού.

3.1 Αρχιτεκτονική συστήματος

Η γενική αρχιτεκτονική του συστήματος έχει στο χαμηλό επίπεδο τη δικτυακή υποδομή μέσω της οποίας επικοινωνούν οι κόμβοι και ανταλλάσσονται τα μηνύματα διαπραγμάτευσης. Ύψηλότερα, με τη χρήση αυτού του επιπέδου, οι συνεργαζόμενοι κόμβοι ανταλλάσσουν μηνύματα διαπραγμάτευσης μέχρι να φτάσουν σε μία συμφωνία για την τελική προσφορά που θα καθορίσει τον τρόπο εκτέλεσης του ερωτήματος προς βελτιστοποίηση. Οι κόμβοι-αγοραστές εντοπίζουν τους κόμβους-πωλητές με την βοήθεια μίας υπηρεσίας χρυσού οδηγού.

3.2 Εξυπρέτης-καταλόγου

Στην παρούσα εργασία υλοποιήθηκε ένας κεντριοποιημένος εξυπρέτης κατάλογος έναντι ενός κατανεμημένου. Για να είναι σε θέση να πληροφορεί τους άλλους κόμβους του συστήματος ποιοι κόμβοι διαθέτουν ποιες σχέσεις του σχήματος της

βάσης διατηρεί μία δομή στη μνήμη με τις απαραίτητες πληροφορίες. Ο κάθε κόμβος που ανήκει στο κατανεμημένο δίκτυο πρέπει να ενημερώνει τον εξυπρέτη για τις αλλαγές που γίνονται στο σχήμα των τοπικών βάσεων δεδομένων όπως για παράδειγμα τη διαγραφή ή την πρόσθεση μιας σχέσης σε ένα κόμβου του δικτύου. Η επικοινωνία μεταξύ των κόμβων γίνεται μέσω της δικτυακής υποδομής με την ανταλλαγή των κατάλληλων μηνυμάτων που έχουν οριστεί.

3.3 Κόμβος-πωλητής

Ένας κόμβος-πωλητής περιορίζει κάθε ερώτημα για την οποία γίνεται αίτηση για προσφορά στις σχέσεις που είναι διαθέσιμες τοπικά. Στη συνέχεια διασπά το αρχικό ερώτημα σε ένα σύνολο ερωτημάτων έτσι ώστε να απαλειφθούν όλα τα Καρτεσιανά γινόμενα καθώς είναι πολύ απαιτητικά από άποψη υπολογιστικού φόρτου. Στη συνέχεια για κάθε ερώτημα του συνόλου αυτού εκτελεί μια παραλλαγή του δυναμικού προγραμματισμού. Ουσιαστικά κατά εκτέλεσή του για κάθε επερώτηση βρίσκεται το βέλτιστο πλάνο πρόσβασης για ένα πίνακα, για ζεύξη δύο πινάκων, ζεύξη τριών πινάκων κλπ. Αυτό είναι χρήσιμο καθώς για έναν κόμβο αγοραστή μπορεί να είναι χρήσιμα μερικά αποτελέσματα οπότε για την όσο το δυνατόν καλύτερη προσέγγιση του βέλτιστου πλάνου γίνεται προσφορά και για αυτά από τους κόμβους-πωλητές.

4 Συμπεράσματα

Στην παρούσα εργασία παρουσιάστηκε μια νέα προσέγγιση στη βελτιστοποίηση ερωτημάτων σε ένα κατανεμημένο δίκτυο αυτόνομων κόμβων. Τα αποτελέσματα πειραματικής αξιολόγησης και σύγκρισης του παρουσιαζόμενου αλγόριθμου με τους δύο άλλους μέχρι τώρα γνωστούς της περιοχής δείχνει την υπεροχή του όσον αφορά την ποιότητα των παραγόμενων πλάνων εκτέλεσης αλλά και την καλύτερη σχέση χρόνου βελτιστοποίησης και ποιότητα παραγόμενου πλάνου εκτέλεσης. Το παραπάνω σε συνδυασμό με το σεβασμό στην αυτονομία των κόμβων, τον καθιστούν πιο κατάλληλο για μεγάλα κατανεμημένα δίκτυα αυτόνομων κόμβων.

Αναφορές

1. Fragkiskos Pentaris, Yannis E. Ioannidis: Query optimization in distributed networks of autonomous database systems. *ACM Trans. Database Syst.* 31(2): 537-583 (2006).
2. Yannis E. Ioannidis, Raymond T. Ng, Kyuseok Shim, Timos K. Sellis: Parametric Query Optimization. *VLDB J.* 6(2): 132-151 (1997).
3. Donald Kossmann, Konrad Stocker: Iterative dynamic programming: a new class of query optimization algorithms. *ACM Trans. Database Syst.* 25(1): 43-82 (2000).
4. Mariposa, "Mariposa Distributed Database Management Systems", User's Manual, <http://s2k-ftp.cs.berkeley.edu:8000/mariposa//src/alpha-1/mariposa-manual.pdf>, 2002.

5. Fragkiskos Pentaris, Yannis E. Ioannidis: Distributed Query Optimization by Query Trading. EDBT 2004: 532-550.
6. Michael Stonebraker, Paul M. Aoki, Robert Devine, Witold Litwin, Michael A. Olson: Mariposa: A New Architecture for Distributed Data. ICDE 1994: 54-65.

Μεθοδολογικό Πλαίσιο για την Ανάπτυξη Εύχρηστων Υπολογιστικών Συστημάτων

Αναστάσιος Σπηλιωτόπουλος

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών,
Τμήμα Πληροφορικής και Τηλεπικοινωνιών¹
tasos@di.uoa.gr

Περίληψη Η συγκεκριμένη εργασία πραγματεύεται την κατασκευή ενός μεθοδολογικού πλαισίου για την ανάπτυξη εύχρηστων υπολογιστικών συστημάτων. Ένα τέτοιο μεθοδολογικό πλαίσιο μπορεί να διευκολύνει σημαντικά τη διαδικασία ανάπτυξης, καθώς επιτρέπει την επιλογή των κατάλληλων μεθόδων ανθρωποκεντρικού σχεδιασμού, ανάλογα με τις ιδιαιτερότητες του συστήματος προς ανάπτυξη σε κάθε περίπτωση.

Λέξεις κλειδιά: ευχρηστία, ανθρωποκεντρικός σχεδιασμός, μεθοδολογικό πλαίσιο, συγκριτική αξιολόγηση, μέθοδος

1 Εισαγωγή

Όλοι οι ορισμοί στη βιβλιογραφία για την ευχρηστία υπολογιστικών συστημάτων (computer systems usability) περιλαμβάνουν την ικανότητα των συστημάτων αυτών να εκπληρώνουν τις προσδοκίες των χρηστών σε ένα συγκεκριμένο πλαίσιο χρήσης. Συνεπώς, για να καταστεί δυνατή η ανάπτυξη εύχρηστων υπολογιστικών συστημάτων απαιτείται η μελέτη τόσο των ιδιαιτεροτήτων των χρηστών που θα τα χρησιμοποιήσουν, όσο και του πλαισίου χρήσης τους. Εν ολίγοις, είναι αναγκαία μία ανθρωποκεντρική προσέγγιση. Για αυτόν το λόγο, η ευχρηστία υπολογιστικών συστημάτων εντάσσεται στην ευρύτερη επιστημονική περιοχή της Επικοινωνίας Ανθρώπου-Υπολογιστή (EAY), αφού αυτή η περιοχή ασχολείται με τη μελέτη και την κατανόηση του ανθρώπου ως χρήστη του υπολογιστή.

Μία τέτοια ανθρωποκεντρική προσέγγιση στην ανάπτυξη υπολογιστικών συστημάτων επιτυγχάνεται με τη χρήση ανθρωποκεντρικών μοντέλων και μεθοδολογιών ανάπτυξης, αλλά και των αντίστοιχων μεθόδων. Οι μέθοδοι αυτές μπορεί να διαφέρουν σημαντικά μεταξύ τους σε πολλές παραμέτρους, ανάλογα με τη φάση της διαδικασίας ανάπτυξης στην οποία χρησιμοποιούνται, το είδος τους, το είδος των αποτελεσμάτων που παρέχουν, το κόστος τους σε χρήματα, χρόνο και ανθρώπινο δυναμικό κ.ά. Για αυτόν το λόγο, είναι επιτακτική η ανάγκη όχι μόνο περιγραφής και ανάλυσης των διαφορετικών αυτών μεθόδων, αλλά και ένταξής τους σε ένα συγκεκριμένο πλαίσιο ανάλογα με τα χαρακτηριστικά τους και τον τρόπο χρήσης τους.

Σκοπός της παρούσης εργασίας είναι η κατασκευή ενός μεθοδολογικού πλαισίου για την ανάπτυξη εύχρηστων υπολογιστικών συστημάτων, το οποίο περιλαμβάνει

¹ Επιβλέπων: Δρακούλης Μαρτάκος

περιγραφή και ανάλυση των σημαντικότερων και πιο ευρέως χρησιμοποιούμενων ανθρωποκεντρικών μεθόδων ανάπτυξης από τη βιβλιογραφία και συγκριτική αξιολόγησή τους βάσει μιας ομάδας συγκεκριμένων παραμέτρων αξιολόγησης, ώστε να καταστούν σαφή τα πλεονεκτήματα, μειονεκτήματα και ιδιαιτερότητες της κάθε μεθόδου. Το αποτέλεσμα είναι η κατάρτιση ενός πίνακα συγκριτικής αξιολόγησης ο οποίος ουσιαστικά αποτελεί ένα μεθοδολογικό πλαίσιο για την ανάπτυξη εύχρηστων υπολογιστικών συστημάτων. Αυτή η συγκριτική αξιολόγηση, σε συνδυασμό με την κατανόηση της λειτουργίας των μεθόδων από την περιγραφή και ανάλυσή τους και την αντίληψη των πραγματικών αναγκών του υπό ανάπτυξη συστήματος, μπορεί να διευκολύνει σημαντικά την επιλογή του κατάλληλου συνδυασμού μεθόδων για την ανάπτυξη ενός εύχρηστου υπολογιστικού συστήματος.

2 Ευχρηστία Υπολογιστικών Συστημάτων

2.1 Επικοινωνία Ανθρώπου-Υπολογιστή

Η επιστημονική περιοχή που είναι γνωστή ως **Επικοινωνία Ανθρώπου-Υπολογιστή** (Human-Computer Interaction ή HCI) ασχολείται με το *σχεδιασμό, την αξιολόγηση και την υλοποίηση διαδραστικών υπολογιστικών συστημάτων* (interactive computer systems) για ανθρώπινη χρήση και με τη μελέτη των σημαντικότερων φαινομένων γύρω από αυτά [1].

Η επικοινωνία του ανθρώπου με τον υπολογιστή μπορεί να περιγραφεί μέσω κάποιων μοντέλων αλληλεπίδρασης, τα οποία αποτελούν το θεωρητικό υπόβαθρο της επιστημονικής περιοχής.

Γνωστικά Μοντέλα

Κατά την αλληλεπίδραση ανθρώπου και υπολογιστή, ο άνθρωπος δέχεται κάποια ερεθίσματα, τα οποία του προκαλούν *γνωστικές διεργασίες*, δηλαδή διεργασίες οι οποίες συνεπάγονται ή αφορούν απόκτηση γνώσης, όπως η κατανόηση, η ενθύμηση, η ανάπτυξη συλλογισμών, η απόκτηση ικανοτήτων, η δημιουργία νέων ιδεών κλπ.[2]. Οι ενέργειες και η γενικότερη συμπεριφορά του ανθρώπου κατά τη διάρκεια ενός τέτοιου είδους αλληλεπίδρασης (δηλαδή μιας αλληλεπίδρασης κατά την οποία κάνει χρήση γνωστικών λειτουργιών) περιγράφεται από *γνωστικά μοντέλα*.

Ένα από τα πιο σημαντικά γνωστικά μοντέλα είναι το **μοντέλο του ανθρώπου ως επεξεργαστή πληροφορίας** ή **μοντέλο ανθρώπινου επεξεργαστή** (human information processing model) [2]. Η κεντρική ιδέα στην οποία στηρίζεται το μοντέλο αυτό είναι ότι η ανθρώπινη συμπεριφορά οδηγείται από μια σειρά σταδίων επεξεργασίας.

Ένα άλλο γνωστικό μοντέλο εξίσου δημοφιλές στο χώρο της ΕΑΥ είναι το **μοντέλο διάδρασης χρήστη-συστήματος κατά Norman** [2] ή αλλιώς το **μοντέλο των επτά σταδίων δράσης** (seven stages of action) [3]. Σύμφωνα με αυτή την προσέγγιση, η διάδραση πραγματοποιείται μέσω ενός διαδοχικά επαναλαμβανόμενου κύκλου ενεργειών, μέχρι την επίτευξη του τελικού στόχου του χρήστη.

Τα προηγούμενα γνωστικά μοντέλα ουσιαστικά προσέφεραν έναν τρόπο για τη σύλληψη και την κατανόηση των αλληλεπιδράσεων μεταξύ ενός χρήστη και ενός συστήματος. Το **Model Human Processor (MHP)** [4] κάνει μια επιπλέον προσπάθεια

για να θέσει τη βάση για την ανάπτυξη ποσοτικών προβλέψεων της απόδοσης του χρήστη και, κατά συνέπεια, τη δημιουργία μεθόδων και εργαλείων για την αξιολόγηση διαφορετικών διεπαφών για συγκεκριμένες εργασίες. Για να επιδειχθεί ο τρόπος με τον οποίο μπορεί να χρησιμοποιηθεί το MHP για την αξιολόγηση διαδραστικών συστημάτων, το μοντέλο αυτό επεκτάθηκε με την ανάπτυξη μιας οικογένειας προβλεπτικών (predictive) μοντέλων, γνωστή ως **GOMS** (Goals, Operators, Methods and Selection rules: Στόχοι, Λειτουργίες, Μέθοδοι και Κανόνες Επιλογής). Συνοπτικά, ένα μοντέλο GOMS είναι μια περιγραφή της γνώσης που χρειάζεται να έχει ένας χρήστης, ώστε να μπορεί να φέρει σε πέρας εργασίες σε ένα διαδραστικό σύστημα [4, 5].

Νοητικά Μοντέλα

Πέρα από τα γνωστικά μοντέλα, τα πιο σημαντικά από τα οποία περιγράφηκαν παραπάνω, για τη μελέτη της αλληλεπίδρασης ανθρώπου-υπολογιστή χρήσιμα μπορούν να αποδειχθούν και τα *νοητικά μοντέλα* (mental models). Ένα νοητικό μοντέλο αποτελεί μια αναπαράσταση γνώσης που χρησιμοποιείται από τον άνθρωπο για την οργάνωση της εμπειρίας του σχετικά με το περιβάλλον και τα αντικείμενα με τα οποία αλληλεπιδρά. Ένα παράδειγμα τέτοιου νοητικού μοντέλου αποτελεί η χρήση μεταφορών στην οθόνη του ηλεκτρονικού υπολογιστή (π.χ. η μεταφορά του περιβάλλοντος γραφείου και η μεταφορά του μενού επιλογών). Τα *ιδεατά μοντέλα* (conceptual models) είναι ένας γενικός όρος που περιγράφει όλα τα νοητικά μοντέλα που εμπλέκονται κατά τις φάσεις σχεδίασης, ανάπτυξης και λειτουργίας ενός υπολογιστικού συστήματος, δηλαδή τους διαφορετικούς τρόπους με τους οποίους το σύστημα γίνεται αντιληπτό από τους σχεδιαστές και τους χρήστες του [2].

Η επιτυχής χρήση ενός διαδραστικού συστήματος (συμπεριλαμβανομένων των υπολογιστικών συστημάτων) απαιτεί να είναι ορατές οι αρχές λειτουργίας του συστήματος, όλες οι ενέργειες να είναι συνεπείς με το ιδεατό μοντέλο και τα ορατά μέρη του συστήματος να αντικατοπτρίζουν την τρέχουσα κατάσταση με έναν τρόπο συνεπή προς το ιδεατό μοντέλο του συστήματος. Έτσι, ο σχεδιαστής πρέπει να αναπτύξει ένα ιδεατό μοντέλο, το οποίο είναι κατάλληλο για τον χρήστη, καταγράφει τα σημαντικά τμήματα της λειτουργίας του συστήματος και είναι εύκολα κατανοητό από τον χρήστη. Ο Norman [3] αναγνωρίζει τρεις διαφορετικές διαστάσεις νοητικών μοντέλων για ένα διαδραστικό σύστημα: το *μοντέλο του σχεδιαστή*, το *μοντέλο του χρήστη* και την *εικόνα του συστήματος*.

2.2 Ορισμός της Ευχρηστίας

Όπως μπορεί να συμπεράνει κανείς μελετώντας τις γνωστικές περιοχές της EAY και τις συσχετίσεις τους [1], κατά τη διαδικασία ανάπτυξης συστημάτων, ιδιαίτερη έμφαση δίδεται στις μετρήσεις της ευχρηστίας είτε κατά τη σχεδίαση είτε κατά τη φάση μέτρησης της ευχρηστίας του τελικού πρωτοτύπου. Έτσι, μία έννοια που τείνει να αποκτήσει κεντρικό ρόλο στο χώρο της EAY είναι η **ευχρηστία συστημάτων** (system usability).

Σύμφωνα με το διεθνές πρότυπο ISO 9241-11 [6], η ευχρηστία ορίζεται ως «η ικανότητα ενός προϊόντος να επιτυγχάνει συγκεκριμένους στόχους αποτελεσματικά,

αποδοτικά και παρέχοντας υποκειμενική ικανοποίηση στους χρήστες του, όταν χρησιμοποιείται σε συγκεκριμένο πλαίσιο χρήσης.»

Ο παραπάνω ορισμός τείνει να γίνει ο κύριος ορισμός αναφοράς της ευχρηστίας συστημάτων [7]. Πέρα από το γεγονός ότι αναγνωρίζεται ως ο καθιερωμένος ορισμός για την ευχρηστία σε έναν συνεχώς αυξανόμενο αριθμό άρθρων της βιβλιογραφίας του χώρου, είναι και ο καθιερωμένος ορισμός του *Common Industry Format* (CIF) για έλεγχο και αξιολόγηση της ευχρηστίας, το οποίο στη συνέχεια έγινε πρότυπο από τον οργανισμό προτυποποίησης ANSI (ANSI/NCITS 354-2001) [8] και εξελίχθηκε σε πρότυπο από τον οργανισμό προτυποποίησης ISO (ISO/IEC 25062:2006) [9], μόλις το 2006. Επιπλέον, το πρότυπο ISO 13407 [10], το οποίο παρέχει οδηγίες για ανθρωποκεντρικό σχεδιασμό κάνει χρήση του παραπάνω ορισμού από το ISO 9241-11.

Ο εξίσου γνωστός και, ενδεχομένως, το ίδιο ευρέως χρησιμοποιούμενος στη βιβλιογραφία ορισμός της ευχρηστίας που έχει δοθεί από τον Nielsen θεωρεί ότι η ευχρηστία ενός συστήματος αναλύεται σε πέντε παραμέτρους [11]:

- *ευκολία εκμάθησης* (learnability). Το σύστημα πρέπει να είναι εύκολο στην εκμάθησή του, έτσι ώστε ο χρήστης να μπορεί να ξεκινήσει την εργασία του γρήγορα.
- *υψηλή απόδοση εκτέλεσης έργου* (efficiency). Το σύστημα πρέπει να είναι αποδοτικό, έτσι ώστε όταν ο χρήστης μάθει τη χρήση του, να μπορεί να επιτύχει ένα υψηλό επίπεδο παραγωγικότητας.
- *ευκολία συγκράτησης της γνώσης* του χρήστη (memorability). Ένας περιστασιακός χρήστης μπορεί να επιστρέψει στο σύστημα και να το χρησιμοποιήσει με επιτυχία χωρίς να πρέπει να το ξαναμάθει από την αρχή, ακόμα και αν έχει πολύ καιρό να το χρησιμοποιήσει.
- *χαμηλή συχνότητα σφαλμάτων* χρήστη. Οι χρήστες του συστήματος πρέπει να υποχρεώνονται σε όσο το δυνατό λιγότερα λάθη κατά τη χρήση του και να μπορούν να ανακάμπτουν εύκολα από αυτά όταν συμβαίνουν. Επιπλέον, το σύστημα δεν πρέπει να επιτρέπει καταστροφικά λάθη.
- *υποκειμενική ικανοποίηση* του χρήστη. Το σύστημα πρέπει να είναι ευχάριστο στη χρήση του, ώστε να προσφέρει υποκειμενική ικανοποίηση στους χρήστες.

Ο ορισμός κατά Nielsen, όντας πιο ανεπίσημος και εμπειρικός, είναι λίγο πιο αναλυτικός και δείχνει να έχει τη μορφή γενικών κατευθυντήριων οδηγιών, ενώ φαίνεται πως μπορεί να αναλυθεί περαιτέρω σε πιο συγκεκριμένες οδηγίες οι οποίες μπορούν να ληφθούν υπόψη από επαγγελματίες του χώρου, όπως αναλυτές, σχεδιαστές και προγραμματιστές συστημάτων.

3 Το Μεθοδολογικό Πλαίσιο

3.1 Εισαγωγή

Επειδή η ευχρηστία ορίζεται μόνο εντός ενός συγκεκριμένου πλαισίου χρήσης, είναι φανερό ότι για την ανάπτυξη εύχρηστων υπολογιστικών συστημάτων απαιτείται μία ανθρωποκεντρική προσέγγιση, δηλαδή μια προσέγγιση στον σχεδιασμό, η οποία δίνει έμφαση στους ανθρώπους που θα χρησιμοποιήσουν το προϊόν [10]. Τα διάφορα

μοντέλα και μεθοδολογίες για ανθρωποκεντρική ανάπτυξη υπολογιστικών συστημάτων κάνουν χρήση ενός εύρους μεθόδων που επιτρέπουν ή διευκολύνουν τον ανθρωποκεντρικό σχεδιασμό.

3.2 Περιγραφή των Μεθόδων

Μελετώντας τις μεθοδολογίες ανάπτυξης και σχεδιασμού υπολογιστικών συστημάτων διαπιστώνουμε ότι οι περισσότερες δραστηριότητες κατά τη διαδικασία ανάπτυξης μπορούν να ταξινομηθούν σε τέσσερις γενικές κατηγορίες: *ανάλυση, σχεδιασμό, υλοποίηση και αξιολόγηση (διαμορφωτική ή συμπερασματική)*. Συνεπώς, θα φαινόταν εκ πρώτης όψεως λογικό να κατηγοριοποιηθούν με τον ίδιο τρόπο και οι μέθοδοι που μπορούν να υποστηρίξουν τις δραστηριότητες αυτές. Όμως, μία τέτοια κατηγοριοποίηση δεν επαρκεί, επειδή δεν επιτρέπει το διαχωρισμό των διαφορετικών ειδών των αποτελεσμάτων που παρέχουν οι μέθοδοι, ενώ, παράλληλα, υπάρχουν και ορισμένες μέθοδοι που χρησιμοποιούνται σε περισσότερες από μία φάσεις της διαδικασίας ανάπτυξης, με αποτέλεσμα μία τέτοια κατηγοριοποίηση ενδεχομένως να προκαλούσε σύγχυση.

Άλλες προσπάθειες κατηγοριοποίησης μεθόδων ευχρηστίας κάνουν προσπάθεια να διακρίνουν *στρατηγικές* για την αξιολόγηση της ευχρηστίας [12, 13], οι οποίες βασίζονται κυρίως στους πόρους σε επίπεδο ανθρώπων και συστήματος. Ωστόσο, ακόμα και σε αυτή την περίπτωση, σε δεύτερο επίπεδο οι μέθοδοι αξιολογούνται σύμφωνα με τον τρόπο χρήσης τους με στόχο την ένταξή τους σε έναν *πίνακα ανάλυσης χρήσης (usage analysis table)*. Με αυτό τον τρόπο, η αρχική κατηγοριοποίηση δεν αντανακλά το ουσιαστικό νόημα της ταξινόμησης των μεθόδων (η οποία γίνεται βάσει τους τρόπου χρήσης τους) και έτσι δεν προσφέρει κάποια διορατικότητα.

Φαίνεται, λοιπόν, πιο δόκιμο να κατηγοριοποιηθούν οι μέθοδοι σε πρώτο επίπεδο ανάλογα με τον τρόπο λειτουργίας τους, δηλαδή τον τρόπο με τον οποίο χρησιμοποιούνται για την εξαγωγή των αποτελεσμάτων. Σε γενικές γραμμές, φαίνεται πως οι μέθοδοι που έχουν διαφορετικό τρόπο λειτουργίας παρουσιάζουν υψηλή συμπληρωματικότητα και μικρή επικάλυψη στα αποτελέσματά τους.

Συνεπώς, οι μέθοδοι, σύμφωνα με τον τρόπο χρήσης τους, διακρίνονται σε τέσσερις κατηγορίες: *διερευνητικές μέθοδοι, μέθοδοι προτυποποίησης, μέθοδοι επιθεώρησης και εμπειρικές μέθοδοι*.

Διερευνητικές Μέθοδοι

Οι διερευνητικές μέθοδοι (inquiry methods) αφορούν τη διερεύνηση των χαρακτηριστικών της ευχρηστίας ενός υπάρχοντος υπολογιστικού συστήματος, ενός συστήματος υπό κατασκευή ή ενός μελλοντικού συστήματος μέσω της καταγραφής των απόψεων των χρηστών και άλλων εμπλεκομένων στο σύστημα. Οι δραστηριότητες των μεθόδων αυτών απαιτούν συνήθως τη συμμετοχή ενός σημαντικού αριθμού χρηστών, είναι δυνατόν, όμως, να πραγματοποιηθούν εκτός κάποιου εργαστηρίου ευχρηστίας (π.χ. στο χώρο εργασίας) και δεν απαιτούν ιδιαίτερο εξοπλισμό. Οι διερευνητικές μέθοδοι μπορούν να χρησιμοποιηθούν είτε νωρίς στη διαδικασία ανάπτυξης, π.χ. στο πλαίσιο της ανάλυσης των απαιτήσεων των χρηστών ή ως μέρος της διαμορφωτικής αξιολόγησης, είτε αργότερα για τη συμπερασματική αξιολόγηση

του συστήματος. Επειδή οι διερευνητικές μέθοδοι βασίζονται στις απόψεις των χρηστών, τα αποτελέσματά τους είναι συνήθως υποκειμενικού χαρακτήρα.

Σε αυτή την κατηγορία περιλαμβάνονται και κάποιες αναλυτικές μέθοδοι, όπως η ανάλυση εργασιών, οι οποίες δεν απαιτούν την άμεση εμπλοκή των χρηστών, αλλά δεν παύουν να έχουν διερευνητικό χαρακτήρα (σε αντίθεση με τις μεθόδους επιθεώρησης). Αυτές οι αναλυτικές μέθοδοι χρησιμοποιούνται συνήθως για την καταγραφή των αποτελεσμάτων που λαμβάνονται από τις πρώτες διερευνητικές μεθόδους.

Παραδείγματα διερευνητικών μεθόδων αποτελούν τα ερωτηματολόγια, οι ατομικές συνεντεύξεις χρηστών, οι συνεντεύξεις πλαισίου χρήσης (contextual interviews), οι ομάδες εργασίας (focus groups), οι εθνογραφικές μελέτες (ethnographic studies), οι παρατηρήσεις πεδίου (field observations), η ταξινόμηση καρτών (card sorting), τα διαγράμματα συγγένειας (affinity diagrams), οι δημοσκοπήσεις, τα ημερολόγια χρηστών, η τεχνική κρίσιμων συμβάντων (critical incidents technique), η ανάλυση εργασιών (task analysis), η κατανομή εργασιών (task allocation), οι επίπλαστες προσωπικότητες (personas), οι περιπτώσεις χρήσης (use cases), και τα σενάρια χρήσης (use case scenarios).

Μέθοδοι Προτυποποίησης

Η προτυποποίηση έχει σκοπό τη μοντελοποίηση του τελικού συστήματος, καθιστώντας δυνατή την εξέταση των χαρακτηριστικών του, ακόμα και όταν αυτό δεν έχει ολοκληρωθεί. Η προτυποποίηση μπορεί να χρησιμοποιηθεί σε οποιοδήποτε στάδιο της διαδικασίας ανάπτυξης. Είναι δυνατό να εφαρμοστεί μια επαναληπτική, εξελικτική προσέγγιση στην προτυποποίηση, ώστε να αναπτύσσονται πρωτότυπα με όλο και περισσότερη λειτουργικότητα και πιστότητα στο τελικό σύστημα. Ένα πρωτότυπο μπορεί να χαρακτηριστεί *υψηλής ή χαμηλής πιστότητας*, ανάλογα με την πιστότητά του στο τελικό σύστημα και το επίπεδο λεπτομέρειας, και *οριζόντιο ή κάθετο*, ανάλογα με το εύρος των λειτουργιών και των χαρακτηριστικών του τελικού συστήματος που περιλαμβάνει. Για παράδειγμα, ένα πρωτότυπο μπορεί να χαρακτηριστεί κάθετο πρωτότυπο υψηλής πιστότητας ή οριζόντιο πρωτότυπο χαμηλής πιστότητας.

Παραδείγματα τέτοιων μεθόδων είναι η κατασκευή πρωτότυπων σε χαρτί (paper prototyping), τα storyboards, η κατασκευή πρωτότυπων με χρήση βίντεο (video prototyping), η γρήγορη κατασκευή πρωτότυπων (rapid prototyping), ο παράλληλος σχεδιασμός, η κατασκευή επαναχρησιμοποιήσιμων ή εξελικτικών πρωτότυπων, η κατασκευή οριζόντιων πρωτότυπων, και η κατασκευή κάθετων πρωτότυπων.

Μέθοδοι Επιθεώρησης

Οι μέθοδοι επιθεώρησης (inspection methods) της ευχρηστίας του συστήματος αφορούν την επιθεώρηση του συστήματος ή ενός πρωτότυπου από έναν ή περισσότερους ειδικούς ευχρηστίας. Η επιθεώρηση αυτή γίνεται συνήθως με τη χρήση κάποιων κανόνων που ακολουθούν οι αξιολογητές, ενώ συχνά οδηγείται από περιπτώσεις χρήσης. Οι μέθοδοι επιθεώρησης χρησιμοποιούνται αρκετά συχνά επειδή δεν απαιτούν τη συμμετοχή χρηστών για τη διεξαγωγή τους, ενώ είναι αρκετά ευέλικτες έχοντας τη δυνατότητα να προσφέρουν ποιοτικά και ποσοτικά αποτελέσματα. Επιπλέον, αυτές οι μέθοδοι μπορούν να χρησιμοποιηθούν για τη συμπερασματική αξιολόγηση του τελικού συστήματος, αλλά και τη διαμορφωτική αξιολόγηση πάνω σε ένα πρώιμο πρωτότυπο και, σε συνδυασμό με την ταχύτητα και

το μικρό κόστος διεξαγωγής τους, χαρίζουν ιδιαίτερη ευελιξία στη διαδικασία ανάπτυξης.

Πέρα από τις παραδοσιακές μεθόδους που πραγματοποιούνται με επισκόπηση (review) του συστήματος από ειδικούς, η επιθεώρηση μπορεί να βασίζεται σε σενάρια χρήσης, εστιάζοντας με αυτό τον τρόπο σε συγκεκριμένες προβληματικές περιοχές του συστήματος. Οι μέθοδοι αυτές που οδηγούνται από σενάρια χρήσης ονομάζονται μέθοδοι περιδιάβατος (walkthrough methods).

Παραδείγματα μεθόδων επιθεώρησης είναι η ευρετική αξιολόγηση (heuristic evaluation), το ευρετικό περιδιάβαση (heuristic walkthrough), η επιθεώρηση από ειδικούς ευχρηστίας (expert review), το περιδιάβαση από ειδικούς ευχρηστίας, η επιθεώρηση με χρήση κατευθυντήριων οδηγιών (guidelines review), το περιδιάβαση με χρήση κατευθυντήριων οδηγιών, το γνωστικό περιδιάβαση (cognitive walkthrough), η πλουραλιστική αξιολόγηση (pluralistic evaluation), η τυπική επιθεώρηση ευχρηστίας (formal usability inspection), η επιθεώρηση συνέπειας (consistency inspection), η επιθεώρηση προτύπων (standards inspection), και η ανάλυση πληκτρολογήσεων (keystroke level analysis).

Εμπειρικές Μέθοδοι

Οι εμπειρικές μέθοδοι (usability testing methods) αφορούν τη διεξαγωγή δοκιμών για την αξιολόγηση του τελικού συστήματος ή ενός πρωτότυπου, με την ενεργό συμμετοχή των χρηστών. Κατά τη διάρκεια μιας συνηθισμένης δοκιμής, ένας χρήστης χρησιμοποιεί το σύστημα για την εκτέλεση ορισμένων εργασιών και η αλληλεπίδρασή του με το σύστημα καταγράφεται και μελετάται αργότερα, με σκοπό την εξαγωγή χρήσιμων συμπερασμάτων για τη βελτίωση του συστήματος. Λόγω της απαίτησης των μεθόδων αυτών για εξοπλισμό καταγραφής της αλληλεπίδρασης, τέτοιες δοκιμές παραδοσιακά λάμβαναν χώρα σε ειδικά εξοπλισμένα εργαστήρια ευχρηστίας (usability labs). Τα τελευταία χρόνια, όμως, οι τεχνολογικές εξελίξεις έχουν επιτρέψει τη διεξαγωγή των δοκιμών και έξω από τα ειδικά εργαστήρια με τη χρήση πιο απλού εξοπλισμού ή αυτοματοποιώντας κάποιες από τις διαδικασίες. Επιπλέον, η τάση είναι προς τη διεξαγωγή σημαντικά φθηνότερων δοκιμών και μάλιστα με λιγότερους χρήστες, ώστε να είναι δυνατή η πραγματοποίηση μεγαλύτερου αριθμού δοκιμών. Με αυτό τον τρόπο, μπορούν να αξιολογηθούν περισσότερες λειτουργίες του συστήματος ή περισσότερα πρωτότυπα, προσφέροντας μεγαλύτερη ευελιξία στη διαδικασία ανάπτυξης.

Παραδείγματα εμπειρικών μεθόδων είναι η μέτρηση απόδοσης (performance measurement), το πρωτόκολλο ομιλούντων υποκειμένων (thinking aloud protocol), η εποικοδομητική αλληλεπίδραση (constructive interaction), η αναδρομική δοκιμή (retrospective testing), η εκπαιδευτική μέθοδος (coaching method), ο «μάγος του Οζ» (wizard of Oz technique), και η αυτόματη καταγραφή χρήσης.

3.3 Συγκριτική Αξιολόγηση των Μεθόδων

Στην ενότητα αυτή επιχειρείται μία ανασκόπηση και συγκριτική αξιολόγηση των μεθόδων που περιγράφηκαν στην προηγούμενη ενότητα, με στόχο την κατασκευή του πίνακα, ο οποίος ουσιαστικά αποτελεί ένα μεθοδολογικό πλαίσιο για την ανάπτυξη εύχρηστων υπολογιστικών συστημάτων. Αυτή η συγκριτική αξιολόγηση, σε

συνδυασμό με την κατανόηση της λειτουργίας των μεθόδων από την περιγραφή και ανάλυσή τους και την αντίληψη των πραγματικών αναγκών του υπό ανάπτυξη συστήματος, μπορεί να διευκολύνει σημαντικά την επιλογή του κατάλληλου συνδυασμού μεθόδων για την ανάπτυξη ενός εύχρηστου υπολογιστικού συστήματος.

Οι παράμετροι με βάση τις οποίες καταρτίζεται ο πίνακας της συγκριτικής αξιολόγησης των μεθόδων επιλέχθηκαν ώστε να διευκολύνουν τόσο τη γενική επισκόπηση (η οποία με τη σειρά της διευκολύνει το συνδυασμό των μεθόδων), παρέχοντας μια γενική εικόνα των μεθόδων, όσο και την επιλογή συγκεκριμένων μεθόδων σε επιμέρους περιπτώσεις, δίνοντας βάρος στα σημεία που εστιάζει η κάθε μέθοδος.

Οι παράμετροι αξιολόγησης επεξηγούνται στη συνέχεια.

- *Είδος μεθόδου.* Η κατηγορία της μεθόδου ανάλογα με τον τρόπο χρήσης της, όπως περιγράφηκε στις προηγούμενες ενότητες. Σύμφωνα με αυτή την κατηγοριοποίηση μια μέθοδος είναι διερευνητική, προτυποποίησης, επιθεώρησης ή εμπειρική.
- *Φάση ανάπτυξης.* Η φάση ή οι φάσεις της διαδικασίας ανάπτυξης που τυπικά χρησιμοποιείται η μέθοδος. Οι φάσεις που αναγνωρίζονται στο πλαίσιο αντιπροσωπεύουν τις γενικότερες κατηγορίες δραστηριοτήτων μιας τυπικής διαδικασίας ανάπτυξης υπολογιστικών συστημάτων και είναι ανάλυση, σχεδιασμός, υλοποίηση και αξιολόγηση (διαμορφωτική ή συμπερασματική).
- *Είδος αποτελεσμάτων (ποιοτικά – ποσοτικά).* Το είδος των αποτελεσμάτων μιας μεθόδου παίζει έναν ιδιαίτερο ρόλο, καθώς μέθοδοι που παρέχουν διαφορετικά είδη αποτελεσμάτων συχνά παρουσιάζουν μεγάλη συμπληρωματικότητα και μικρή επικάλυψη καθιστώντας τη διαδικασία ανάπτυξης πιο αποτελεσματική. Τα ποσοτικά αποτελέσματα από τη χρήση μιας μεθόδου μπορούν να αναλυθούν στατιστικά, να παρουσιαστούν και να ερμηνευθούν εύκολα και να χρησιμοποιηθούν για σύγκριση προϊόντων ή ιδεών. Αντίθετα, τα ποιοτικά αποτελέσματα δεν προσφέρονται για εύκολη τεκμηρίωση των ευρημάτων, αλλά μπορεί να αποδειχθούν πολύτιμα καθώς σχετίζονται με έννοιες που δεν μπορούν να ποσοτικοποιηθούν, όπως οι προτιμήσεις και η συμπεριφορά των χρηστών
- *Μεροληψία (υποκειμενικά – αντικειμενικά αποτελέσματα).* Η παράμετρος αυτή αφορά το επίπεδο υποκειμενικότητας ή αντικειμενικότητας των αποτελεσμάτων μιας μεθόδου. Τα αποτελέσματα μιας μεθόδου είναι δυνατό να επηρεαστούν σε μεγάλο βαθμό από τη μεροληψία ή προκατάληψη των χρηστών ή των αξιολογητών.
- *Κόστος της μεθόδου.* Το κόστος για τη χρήση μιας μεθόδου μπορεί να περιλαμβάνει τον απαιτούμενο εξοπλισμό (π.χ. εργαστήριο ευχρηστίας), την υλοποίηση ενός πρωτοτύπου, την απασχόληση χρηστών, ειδικών ευχρηστίας και άλλων εμπλεκόμενων, ταξίδια κλπ.
- *Ανάγκη ειδικών ευχρηστίας.* Η παράμετρος αυτή εξετάζει αν είναι απαραίτητη η απασχόληση ειδικών ευχρηστίας για τη διεξαγωγή ή οργάνωση της δραστηριότητας που περιγράφει η μέθοδος.
- *Αριθμός χρηστών.* Ένας προτεινόμενος τυπικός αριθμός χρηστών για τη μέθοδο, στην περίπτωση που είναι απαραίτητη η απασχόληση χρηστών.
- *Επίπεδο πληροφορίας αποτελεσμάτων (χαμηλό – υψηλό).* Η παράμετρος αυτή αφορά το επίπεδο λεπτομέρειας των πληροφοριών που λαμβάνουμε από τα αποτελέσματα της μεθόδου. Μία μέθοδος είναι δυνατό να δίνει αποτελέσματα διαφορετικών επιπέδων λεπτομέρειας.

- *Αμεσότητα αποτελεσμάτων.* Αυτή η παράμετρος σχετίζεται με την ταχύτητα με την οποία λαμβάνονται τα αποτελέσματα μιας μεθόδου.
- *Χώρος.* Ο χώρος στον οποίο πραγματοποιείται η δραστηριότητα της μεθόδου (π.χ. εργαστήριο ευχρηστίας, χώρος εργασίας).
- *Διακριτικότητα (non intrusiveness).* Η συμπεριφορά ενός χρήστη είναι δυνατό να επηρεαστεί από την παρουσία του παρατηρητή, του ατόμου που παίρνει συνέντευξη ή του εξοπλισμού καταγραφής, σε τέτοιο βαθμό ώστε τα ευρήματα να μην είναι ακριβή.
- *Κύρια πλεονεκτήματα.* Παρουσιάζονται επιγραμματικά τα σημαντικότερα πλεονεκτήματα της μεθόδου. Τα κύρια πλεονεκτήματα της μεθόδου, μαζί με τα κύρια μειονεκτήματά της, διευκολύνουν τη γενική επισκόπηση των μεθόδων, παρέχοντας μια γενική εικόνα για την καθεμία από αυτές και καθιστώντας πιο εύκολο τον αποτελεσματικό συνδυασμό τους.
- *Κύρια μειονεκτήματα.* Παρουσιάζονται επιγραμματικά τα σημαντικότερα μειονεκτήματα της μεθόδου. Τα κύρια μειονεκτήματα της μεθόδου, μαζί με τα κύρια πλεονεκτήματά της, διευκολύνουν τη γενική επισκόπηση των μεθόδων, παρέχοντας μια γενική εικόνα για την καθεμία από αυτές και καθιστώντας πιο εύκολο τον αποτελεσματικό συνδυασμό τους.

3.4 Ο Πίνακας Συγκριτικής Αξιολόγησης

Παρακάτω παρατίθεται ένα τμήμα του πίνακα συγκριτικής αξιολόγησης των μεθόδων ευχρηστίας για τις 4 πρώτες διερευνητικές μεθόδους, για την καλύτερη επίδειξη του μεθοδολογικού πλαισίου.

Πίνακας 1. Απόσπασμα από τον πίνακα συγκριτικής αξιολόγησης μεθόδων για την ανάπτυξη εύχρηστων υπολογιστικών συστημάτων.

| Όνομα μεθόδου | Ομάδες εργασίας | Ατομικές συνεντεύξεις | Συνεντεύξεις πλαισίου χρήσης | Ταξινόμηση καρτών |
|----------------------------------|---|------------------------------------|------------------------------|--------------------------------|
| Είδος μεθόδου | Διερευνητική | Διερευνητική | Διερευνητική | Διερευνητική |
| Φάση ανάπτυξης | Ανάλυση, σχεδιασμός, διαμορφωτική αξιολόγηση. | Ανάλυση, συμπερασματική αξιολόγηση | Ανάλυση απαιτήσεων | Ανάλυση απαιτήσεων, σχεδιασμός |
| Ποιοτικά – ποσοτικά αποτελέσματα | Ποιοτικά | Ποιοτικά | Ποιοτικά | Ποσοτικά |
| Υποκ. – αντικ. αποτελέσματα | Υποκειμενικά | Υποκειμενικά | Υποκειμενικά | Υποκειμενικά |
| Κόστος | Χαμηλό | Χαμηλό | Χαμηλό | Χαμηλό |
| Ανάγκη ειδικών ευχρηστίας | Ναι | Ναι | Ναι | Όχι |
| Αριθμός χρηστών | 6-9 | 5-10 | 5-10 | 5-20 |
| Επίπεδο πληροφορίας | Υψηλό, χαμηλό | Υψηλό | Υψηλό, χαμηλό | Υψηλό, χαμηλό |
| Αμεσότητα αποτελεσμάτων | Ναι | Ναι | Ναι | Όχι |

| Χώρος | Εργαστήριο, χώρος εργασίας | Εργαστήριο, χώρος εργασίας | Χώρος εργασίας | Εργαστήριο, χώρος εργασίας |
|----------------------------|---|---|--|--|
| Διακριτικότητα | Όχι | Όχι | Όχι | Ναι |
| Κύρια πλεονεκτήματα | Οι προτιμήσεις των χρηστών προκύπτουν από αυθόρμητες αντιδράσεις και χρησιμοποιείται η δυναμική της ομάδας. Εύκολα επαναλαμβανόμενο. Μπορεί να εστιάσει σε συγκεκριμένα ζητήματα. | Προσφέρει βαθύτερη κατανόηση των απόψεων και εμπειριών των χρηστών. Ευέλικτη μέθοδος. Χρησιμοποιείται αποτελεσματικά ως συμπληρωματική μέθοδος. | Χρησιμοποιείται αποτελεσματικά ως συμπληρωματική μέθοδος. | Ιδιαίτερα οικονομική μέθοδος που παράγει ποσοτικά αποτελέσματα. Μπορεί να πραγματοποιηθεί από απόσταση με ηλεκτρονικό τρόπο. Χρησιμοποιείται αποτελεσματικά για την οργάνωση του περιεχομένου. |
| Κύρια μειονεκτήματα | Καθαρά υποκειμενικά αποτελέσματα, που δεν είναι εύκολο να τεκμηριωθούν ή να αναλυθούν. | Καθαρά υποκειμενικά αποτελέσματα, που δεν είναι εύκολο να τεκμηριωθούν ή να αναλυθούν. | Καθαρά υποκειμενικά αποτελέσματα, που δεν είναι εύκολο να τεκμηριωθούν ή να αναλυθούν. | Υποκειμενικά αποτελέσματα, απαιτείται στατιστική επεξεργασία των αποτελεσμάτων. |

Αναφορές

1. T. T. Hewett, Baecker, R., Card, S., Carey, T., Gasen, J., Mantei, M., Perlman, G., Strong, G., Verplank, W.: ACM SIGCHI Curricula for Human-Computer Interaction. The Association for Computing Machinery (1992)
2. Ν. Αβούρης: Εισαγωγή στην Επικοινωνία Ανθρώπου-Υπολογιστή. Διάλογος, Αθήνα (2000)
3. D. A. Norman: The Design of Everyday Things. MIT Press, London (1988)
4. S. K. Card, T. P. Moran, and A. Newel: The Psychology of Human-Computer Interaction. Lawrence Erlbaum (1983)
5. D. Kieras: A Guide to GOMS Task Analysis. University of Michigan (1994)
6. ISO: ISO 9241-11:1998 Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs) -- Part 11: Guidance on Usability. ISO (1998)
7. T. Jokela, N. Iivari, J. Matero, and M. Karukka: The Standard of User-Centered Design and the Standard Definition of Usability: Analyzing ISO 13407 against ISO 9241-11. In: 1st Latin American Conference on Human-Computer Interaction, CLIHC 2003, Rio de Janeiro, Brasil (2003)
8. ANSI: Common Industry Format for Usability Test Reports. ANSI/NCITS 354-2001 (2001)
9. ISO/IEC: ISO/IEC 25062:2006 Common Industry Format (CIF) for Usability Test Reports. ISO (2006)
10. ISO: ISO 13407:1999 Human-Centred Design Processes for Interactive Systems. ISO (1999)
11. J. Nielsen: Usability Engineering. Academic Press, San Diego (1993)
12. R. Fitzpatrick and A. Dix: A Process for Appraising Commercial Usability Evaluation Methods. HCI International '99, Munich (1999)
13. R. Fitzpatrick: Strategies for Evaluating Software Usability. Report, Department of Mathematics, Statistics and Computer Science, Dublin Institute of Technology, Ireland (1999)

Πτυχιακές Εργασίες

123

Σχεδίαση και Υλοποίηση Μηχανισμού Αποθήκευσης Περιγραφών Υπηρεσιών Ιστού

Κορμαρής Ιωάννης¹

Πανεπιστήμιο Αθηνών, Τμήμα Πληροφορικής και Τηλεπικοινωνιών
Πανεπιστημιούπολη, Ιλίσια, 15784, Αθήνα, Ελλάς
std02063@di.uoa.gr

Περίληψη. Σε αυτή την πτυχιακή εργασία παρουσιάζεται ο σχεδιασμός και η υλοποίηση μιας μονάδας λογισμικού που αποθηκεύει περιγραφές υπηρεσιών ιστού σε μία σχεσιακή βάση δεδομένων. Ο μηχανισμός αναλαμβάνει να υλοποιήσει το έργο της προσωρινής αποθήκευσης (caching) σε μία μηχανή αναζήτησης υπηρεσιών ιστού. Η ανάγκη για την ενσωμάτωση ενός τέτοιου μηχανισμού, δημιουργείται κυρίως από την ανάγκη εξάλειψης προβλημάτων που οφείλονται στο δίκτυο, αλλά και για λόγους βελτίωσης της απόδοσης της μηχανής. Ορισμένα από τα βασικά θέματα που αντιμετωπίζονται είναι η «αφαίρεση» των απαραίτητων πληροφοριών από ένα αρχείο WSDL, η μετατροπή του XML Σχήματος σε σχεσιακή βάση δεδομένων και ο χειρισμός των επεκτάσεων της γλώσσας WSDL.

Λέξεις-κλειδιά: υπηρεσίες ιστού, WSDL, μηχανισμός προσωρινής αποθήκευσης, μηχανή αναζήτησης, προσωρινή αποθήκευση.

1 Εισαγωγή

Τα τελευταία χρόνια, η ανάπτυξη των τεχνολογιών του Διαδικτύου προσανατολίζεται στις υπηρεσίες. Το πέρασμα από το ανθρωποκεντρικό Διαδίκτυο προς το υπηρεσιοστραφές αποτελεί μία προσπάθεια για εκλέπτυνση και βελτίωση του τρόπου, αλλά και του χρόνου που απαιτείται για τη δημιουργία δικτυακών εφαρμογών. Οι *υπηρεσίες ιστού (web services)*, που αποτελούν μία συλλογή από συναρτήσεις, οι οποίες παρέχονται με τη μορφή ενός ενιαίου πακέτου μέσω του Διαδικτύου, αποτελούν το αποκορύφωμα της προσπάθειας αυτής. Η χρήση της τεχνολογίας αυτής, πέρα από το να παρέχει τις κατάλληλες υποδομές για την οικοδόμηση καταμεμημένων συστημάτων, επιτρέπει στις εταιρίες, αλλά και στον καθένα να μοιράζεται γρήγορα και οικονομικά τις υπηρεσίες του μέσω του Διαδικτύου.

¹ **Επιβλέπουσα:** Αφροδίτη Τσαγκατίδου, Επίκουρος Καθηγήτρια ΕΚΠΑ

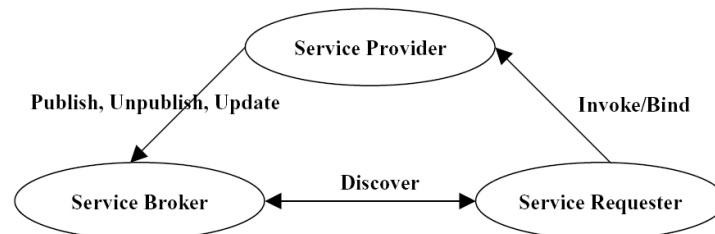
Οι υπηρεσίες ιστού αποτελούν το επόμενο βήμα «αφαίρεσης» μετά από τον αντικειμενοστραφή προγραμματισμό. Κάποια από τα πλεονεκτήματα που τις χαρακτηρίζουν είναι η διαλειτουργικότητα, η ευελιξία και η επαναχρησιμοποίηση.

Καθώς αυξάνονται τα οφέλη που απορρέουν από τη χρήση των υπηρεσιών ιστού, αυξάνεται και ο όγκος τους στο Διαδίκτυο. Παράλληλα εμφανίζεται και η ανάγκη των χρηστών για δυνατότητα εύρεσης των υπηρεσιών ιστού που επιθυμούν, με αποδοτικό τρόπο. Οι αντίστοιχες μηχανές αναζήτησης είναι οι μονάδες λογισμικού που αναλαμβάνουν να καλύψουν αυτή την ανάγκη. Το έργο των μηχανών αναζήτησης δυσχεραίνουν οι καθυστερήσεις του δικτύου και το μεγάλο πλήθος των μητρώων στα οποία αποθηκεύονται οι περιγραφές των υπηρεσιών ιστού.

Η πτυχιακή εργασία αναλαμβάνει να σχεδιάσει και να υλοποιήσει έναν μηχανισμό αποθήκευσης περιγραφών υπηρεσιών ιστού σε τοπικό επίπεδο, ώστε να είναι άμεσα εκμεταλλεύσιμος από μια μηχανή αναζήτησης. Η υλοποίηση του μηχανισμού στοχεύει στη βελτίωση της απόδοσης αλλά και της ποιότητας των αποτελεσμάτων της μηχανής αναζήτησης.

2 Επισκόπηση Τεχνολογίας Υπηρεσιών Ιστού

Οι υπηρεσίες ιστού παρέχουν έναν τυποποιημένο τρόπο επικοινωνίας μεταξύ εφαρμογών ανεξάρτητα από την πλατφόρμα λογισμικού που αυτές εκτελούνται. Το μοντέλο των υπηρεσιών ιστού ακολουθεί το μοτίβο δημοσίευση (*publish / unpublish / update*), εύρεση (*discover*) και σύνδεση (*invoke / bind*) [2], [10].



Σχήμα 1 Το Μοντέλο Υπηρεσιών Ιστού

Στο παραπάνω σχήμα (Σχήμα 1) διακρίνονται οι εξής ρόλοι:

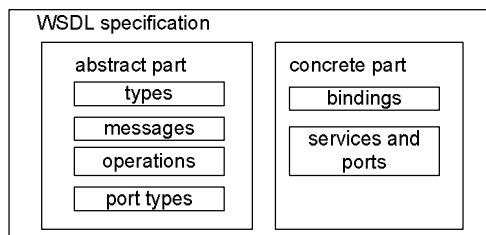
Προμηθευτής υπηρεσίας (Service Provider): Ο προμηθευτής της υπηρεσίας είναι το μέλος που προσφέρει το απαραίτητο λογισμικό και είναι υπεύθυνος για την δημοσίευση (*publish*), απομάκρυνση (*unpublish*) και ενημέρωση (*update*) των υπηρεσιών του. Από επαγγελματική σκοπιά, ο προμηθευτής είναι ο ιδιοκτήτης της υπηρεσίας. Από αρχιτεκτονική σκοπιά, είναι η πλατφόρμα που διαθέτει την υλοποίηση της υπηρεσίας [2], [10].

Πελάτης (Service Requester): Ο πελάτης είναι το μέλος που έχει ανάγκη, η οποία μπορεί να καλυφθεί πλήρως από ένα web service, το οποίο είναι διαθέσιμο στο Διαδίκτυο. Από επαγγελματική σκοπιά, είναι η επιχείρηση ή ο ιδιώτης που απαιτεί την επίτευξη μιας συγκεκριμένης λειτουργίας. Από αρχιτεκτονική σκοπιά, είναι η

εφαρμογή που ζητάει να επικαλεστεί (*invoke*) μία υπηρεσία ιστού. Ο πελάτης μπορεί να είναι ένας άνθρωπος που έχει πρόσβαση στην υπηρεσία μέσω ενός υπολογιστή ή ενός ασύρματου φυλλομετρητή, μία εφαρμογή ή ακόμα και ένα άλλο web service. Ένας πελάτης εντοπίζει την υπηρεσία μέσω ενός μεσάζοντα (*broker*) και συνδέεται στις υπηρεσίες μέσω του προμηθευτή (*service provider*) [2], [10].

Μεσάζων (Service broker): Αυτό το μέλος παρέχει έναν κατάλογο περιγραφών υπηρεσιών, όπου οι προμηθευτές μπορούν να δημοσιεύσουν τις υπηρεσίες τους και οι πελάτες να εντοπίσουν και να αποκτήσουν πληροφορίες για τις δημοσιευμένες υπηρεσίες. Αντιστοιχούν στις τηλεφωνικές λευκές σελίδες. Τέτοια παραδείγματα μεσαζόντων είναι τα μητρώα UDDI (UDDI registries) [2], [10].

Οι τεχνολογίες που συμμετέχουν στο παραπάνω μοντέλο είναι · το πρωτόκολλο SOAP (*Simple Object Access Protocol*) [17] που χρησιμοποιείται για να στείλουμε μηνύματα σε διαφορετικές ενότητες λογισμικού, το UDDI (*Universal Description, Discovery and Integration*) [2] το οποίο χρησιμοποιείται για την βάση αναζήτησης των υπηρεσιών ιστού και τέλος η γλώσσα WSDL (*Web Services Definition Language*) [1] με την οποία περιγράφεται η λειτουργικότητα των υπηρεσιών ιστού.



Σχήμα 2 Δομή εγγράφων WSDL

Η εργασία αυτή μελετάει σε βάθος τη γλώσσα WSDL, η δομή της οποίας φαίνεται στο παραπάνω σχήμα (Σχήμα 2). Σε γενικές γραμμές, ένα έγγραφο WSDL περιγράφει μία ή περισσότερες υπηρεσίες (services) που προσφέρει ένας οργανισμός. Καθεμία από τις υπηρεσίες αυτές προσφέρουν ένα σύνολο από λειτουργίες (operations). Η κάθε λειτουργία ανταλλάσσει μηνύματα (messages) με τον πελάτη. Το κάθε μήνυμα αποτελείται από πολλά μέρη (parts). Τα μέρη αντιστοιχίζονται με τις παραμέτρους μιας συνάρτησης στον προγραμματισμό και επομένως περιγράφουν τον τύπο και το όνομα των παραμέτρων των μηνυμάτων.

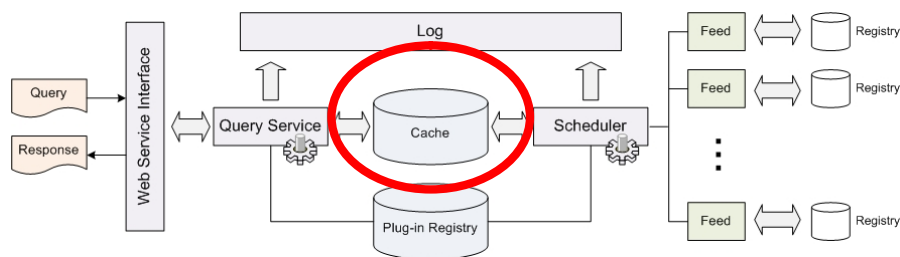
Ο ορισμός των λειτουργιών και της δομής των μηνυμάτων αποτελούν το αφηρημένο (*abstract*) μέρος του εγγράφου WSDL και χρησιμοποιούνται κατά την αναζήτηση των υπηρεσιών. Στο *concrete* (*συγκεκριμένο*) μέρος του WSDL περιγράφονται οι συνδέσεις (bindings) των λειτουργιών σε συγκεκριμένα πρωτόκολλα επικοινωνίας μέσω διαφορετικών θυρών (ports). Οι πληροφορίες αυτές περιγράφουν δηλαδή τον τρόπο με τον οποίο θα κληθούν οι υπηρεσίες που προσφέρει ο συγκεκριμένος προμηθευτής.

Η εργασία αυτή προσφέρει έναν μηχανισμό για την ανάγνωση των πληροφοριών που βρίσκονται στο αφηρημένο μέρος ενός αρχείου WSDL και την κατάλληλη αποθήκευσή τους, ώστε να χρησιμοποιηθούν στη συνέχεια από την μηχανή αναζήτησης υπηρεσιών ιστού USQL Engine, που παρουσιάζεται συνοπτικά στην επόμενη παράγραφο.

3 Η Μηχανή Αναζήτησης Υπηρεσιών Ιστού

Οι περισσότερες μηχανές αναζήτησης, όπως για παράδειγμα το γνωστό Google, χρησιμοποιούν κάποιο μηχανισμό μόνιμης αποθήκευσης των πληροφοριών που αναζητούν, προκειμένου να βελτιώσουν την ταχύτητα ανάκτησης και απόκρισής τους. Στην περίπτωση της αναζήτησης υπηρεσιών ιστού, ένας τέτοιος μηχανισμός μόνιμης αποθήκευσης θα πρέπει να αποθηκεύει κατάλληλα τις περιγραφές των υπηρεσιών ιστού.

Ο μηχανισμός αποθήκευσης υπηρεσιών ιστού που έχει υλοποιηθεί στα πλαίσια αυτής της πτυχιακής εργασίας αποτελεί μέρος τη μηχανής αναζήτησης υπηρεσιών ιστού *USQL engine* [16], η οποία έχει υλοποιηθεί από το εργαστήριο s3lab του Τμήματος Πληροφορικής και Τηλεπικοινωνιών του Πανεπιστημίου Αθηνών. Στο εξής, ο μηχανισμός αποθήκευσης που έχει υλοποιηθεί θα αναφέρεται ως *cache*. Η βάση δεδομένων *cache* αποτελεί ένα πολύ σημαντικό κομμάτι της μηχανής αναζήτησης. Όπως φαίνεται στο παρακάτω σχήμα (**Σχήμα 3**), η *cache* χρησιμοποιείται από τη μηχανή αναζήτησης προκειμένου να εκτελούνται συγκεκριμένες υπηρεσίες επερωτήσεων.



Σχήμα 3 Η μηχανή αναζήτησης υπηρεσιών ιστού USQL

Τα στοιχεία του διαγράμματος που ονομάζονται Feeds είναι υπεύθυνα για την ανάγνωση των μητρώων αποθήκευσης υπηρεσιών ιστού. Υπάρχει διαφορετικό Feed για κάθε υλοποίηση των μητρώων (π.χ. UDDI), ώστε η *cache* να είναι ανεξάρτητη από την υλοποίηση τους. Στον scheduler έχει οριστεί μια καθορισμένη συχνότητα ελέγχου της τοπικής βάσης δεδομένων *cache*. Για παράδειγμα, ο scheduler ελέγχει ανά τρεις μέρες τα μητρώα με τη βοήθεια των Feeds και ενημερώνει την *cache* για τις όποιες αλλαγές.

3.1 Συνεισφορά του Μηχανισμού Προσωρινής Αποθήκευσης (Cache)

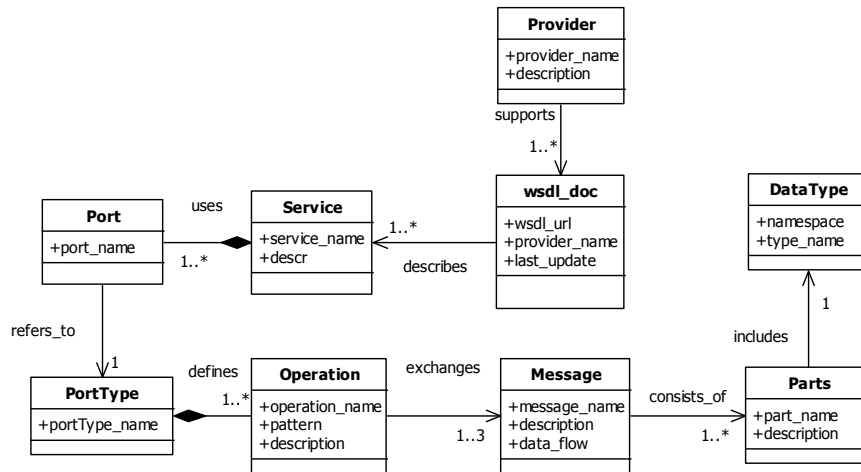
Ο μηχανισμός μόνιμης αποθήκευσης προσφέρει σημαντικά πλεονεκτήματα στην παραπάνω μηχανή αναζήτησης:

- *ταχύτητα*: πιο γρήγορη αναζήτηση εξαιτίας της εξάλειψης καθυστερήσεων δικτύου
- *ανεξαρτησία*: η μηχανή αναζήτησης επιστρέφει αποτελέσματα ακόμα και αν τα μητρώα UDDI δεν είναι προσωρινά διαθέσιμα.
- *μείωση φορτίου*: οι εξυπηρετητές των μητρώων δεν επιβαρύνονται με την αναζήτηση, καθότι η τελευταία πραγματοποιείται σε τοπικό επίπεδο.

- *μείωση όγκου δεδομένων*: αποθήκευση μόνο πληροφοριών που βρίσκονται στο abstract μέρος του αρχείου WSDL.

4 Σχεδιασμός της Βάσης Δεδομένων

Η βάση δεδομένων της εφαρμογής απαιτεί συγκεκριμένες πληροφορίες που αφορούν αποκλειστικά στο abstract μέρος της περιγραφής και επομένως η περιγραφή των υπηρεσιών ιστού στην βάση είναι διαφορετική από αυτήν της WSDL. Ωστόσο οι σχέσεις του XML σχήματος μεταξύ των στοιχείων που αποθηκεύονται παραμένουν, όπως απεικονίζει και το ακόλουθο διάγραμμα (Σχήμα 4).



Σχήμα 4 Αναπαράσταση του σχήματος της βάσης δεδομένων

4.1 Περιορισμοί Αναφορικής Ακεραιότητας

Στο σχήμα της βάσης παρουσιάζεται αναφορικός περιορισμός, εξαιτίας της ύπαρξης ξένων κλειδιών. Σε λογικό επίπεδο, ο περιορισμός αυτός αφορά κυρίως στο γεγονός ότι μία περιγραφή WSDL είναι εξαρτημένη από έναν provider και τα στοιχεία μιας περιγραφής είναι εξαρτημένα από την ίδια την περιγραφή. Για παράδειγμα, έστω ότι ένας provider για κάποιο λόγο σταματήσει να λειτουργεί. Τότε παύουν να ισχύουν και όποιες υπηρεσίες ιστού προσέφερε. Με αντίστοιχο τρόπο, όταν μια υπηρεσία ιστού σταματήσει να διατίθεται, τότε η περιγραφή της δεν υπάρχει πια. Τα παραπάνω παραδείγματα καταδεικνύουν την ανάγκη να αντιμετωπιστεί η αναφορικότητα της βάσης ώστε να αποφευχθεί η συσσώρευση υπολειμμάτων σε περίπτωση που διαγραφούν δεδομένα.

Η υλοποίηση της βάσης αντιμετωπίζει την ύπαρξη τέτοιων περιπτώσεων με τη χρήση αναφορικά πυροδοτούμενων ενεργειών. Συγκεκριμένα επιλέγεται η πράξη της

διαγραφής και της ενημέρωσης να γίνεται CASCADE και στους υπόλοιπους πίνακες στους οποίους υπάρχει αναφορά εξαιτίας των ξένων κλειδιών.

4.2 Επεκτασιμότητα

Ένα σημαντικό χαρακτηριστικό του WSDL είναι η επεκτασιμότητα. Όπως φαίνεται από τις προηγούμενες ενότητες τα δεδομένα αποθηκεύονται σε μία φαινομενικά «στατική» σχεσιακή βάση δεδομένων. Ωστόσο η επεκτασιμότητα της βάσης αποτελεί γεγονός από τη στιγμή που και το WSDL είναι επεκτάσιμο.

Στο τελικό σχήμα της βάσης, πέρα από τους πίνακες της εικόνας 4, έχει προστεθεί επιπλέον ο πίνακας extensions, ο οποίος χρησιμοποιείται για την αποθήκευση των extensibility elements και attributes του εγγράφου WSDL.

Το πεδίο *parser_class* του πίνακα extensions αποθηκεύει ένα όνομα τύπου string, το οποίο αντιστοιχεί στον parser του στοιχείου (element) με όνομα *extension_name* και χώρο ονομάτων *namespace* που βρίσκεται κάτω από το στοιχείο parent. Με παρόμοιο τρόπο τα παραπάνω πεδία μπορεί να χρησιμοποιηθούν και στην περίπτωση που πρόκειται για extensibility attribute. Ένα παράδειγμα για ένα extensibility element φαίνεται στην παρακάτω εικόνα (Σχήμα 5).

| extensions | | | |
|------------|---------------------------------------|----------------|-------------------|
| parent | namespace | extension_name | parser_class |
| port | http://schemas.xmlsoap.org/wsdl/soap/ | address | SoapAddressParser |

Σχήμα 5 Στιγμιότυπο του πίνακα extensions

Ο πίνακας extensions ορίζει ότι τα στοιχεία που περιγράφει μπορούν να διαβαστούν (parsing) από τον μηχανισμό και επομένως να αποθηκευτούν σε κάποια επέκταση της βάσης αυτής (προσθήκη ενός νέου πίνακα) ή σε μία τελείως διαφορετική βάση, στην οποία αποθηκεύονται μόνο επεκτάσεις. Ο χρήστης έχει πλήρη ελευθερία να επιλέξει αυτό που εφαρμόζει καλύτερα στις ανάγκες του. Στην παράγραφο 6.2 περιγράφεται ο τρόπος χειρισμού των επεκτάσεων σε επίπεδο εφαρμογής.

4.3 Σύγκριση XML και SQL Βάσεων Δεδομένων

Η WSDL χρησιμοποιεί μία XML γραμματική για να περιγράψει τις υπηρεσίες ιστού. Θα μπορούσε λοιπόν, να αναρωτηθεί κανείς, για ποιον λόγο είναι απαραίτητο κάθε φορά να εξάγουμε δεδομένα από ένα έγγραφο XML και μετά να τα αποθηκεύουμε σε έναν SQL server. Ακόμα περισσότερο γιατί είναι αναγκαίο να εξάγουμε δεδομένα από έναν SQL server και ύστερα να τα διαμορφώνουμε σε XML.

Η χρήση μιας XML βάσης δεδομένων θα συντελούσε στο να αποφύγει κανείς τη διαδικασία που περιγράφηκε παραπάνω. Όμως, συγκεκριμένα προβλήματα κάνουν τις XML βάσεις να αδυνατούν έναντι των δημοφιλών SQL βάσεων δεδομένων.

Η τεχνολογία στις σχεσιακές βάσεις δεδομένων αναπτύσσεται εδώ και πολλά χρόνια και επομένως βρίσκεται αρκετά βήματα μπροστά από τις βάσεις XML. Είναι ιδιαίτερα δημοφιλής στους περισσότερους χρήστες και επομένως καθιστά το

περιβάλλον της εφαρμογής πιο «φιλικό». Οι SQL βάσεις αποθηκεύουν την απαραίτητη πληροφορία εξοικονομώντας χώρο σε αντίθεση με την XML, όπου πολλές φορές είναι αναγκαίο να διατηρούνται και αρκετά μεταδεδομένα. Οι επερωτήσεις σε μία XML βάση απαιτούν την πρόσθετη διαδικασία του parsing αυξάνοντας την πολυπλοκότητα και μειώνοντας αισθητά την απόδοση, όταν ο όγκος των δεδομένων είναι ιδιαίτερα μεγάλος. Πολύ μεγάλα αρχεία είναι ιδιαίτερα δύσκολο να αποθηκευτούν καθότι απαιτείται streaming. Τέλος δεν μπορούν να πραγματοποιηθούν οι μερικές ενημερώσεις (partial updates).

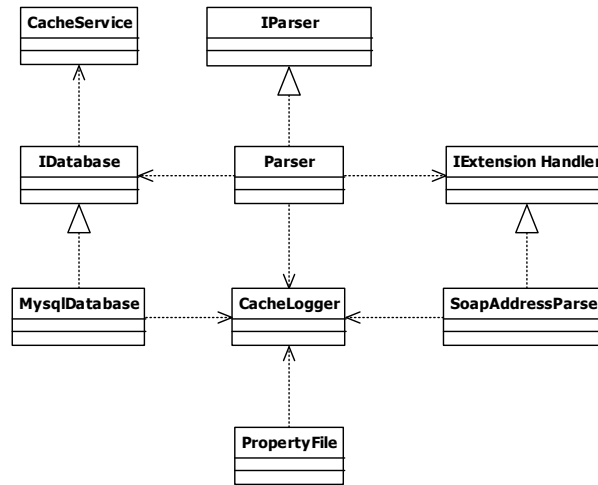
Για όλους τους παραπάνω λόγους επιλέχτηκε τελικά η χρήση μιας SQL βάσης δεδομένων.

5 Ανάλυση και Σχεδίαση του Μηχανισμού Αποθήκευσης

Ο απότερος στόχος του μηχανισμού είναι η εισαγωγή των δεδομένων σε μια σχεσιακή βάση δεδομένων, όπως αυτή περιγράφηκε στην προηγούμενη ενότητα. Επιπλέον, ο μηχανισμός πρέπει να παρέχει την κατάλληλη προγραμματιστική διεπαφή (API) προκειμένου να μπορέσει να ενταχθεί στη μηχανή αναζήτησης που περιγράφηκε στην παράγραφο 3. Η λειτουργικότητα του μηχανισμού συνοψίζεται στο ότι πρέπει να δέχεται στην είσοδό του ένα έγγραφο WSDL, να εξάγει τις απαραίτητες πληροφορίες και να τις αποθηκεύει στη βάση.

5.1 Αρχιτεκτονική

Το ακόλουθο διάγραμμα κλάσεων (**Σχήμα 6**) απεικονίζει τα βασικά συστατικά του μηχανισμού αποθήκευσης περιγραφών υπηρεσιών ιστού. Η εφαρμογή αποτελείται από τρεις βασικές διεπαφές. Η διεπαφή *IDatabase* ορίζει το σύνολο των πράξεων που μπορούν να πραγματοποιηθούν πάνω στην βάση δεδομένων της cache. Στην υλοποίηση που πραγματοποιήθηκε επιλέχτηκε η διασύνδεση με την σχεσιακή βάση δεδομένων MySQL μέσω της κλάσης *MysqlDatabase*. Η διεπαφή *IExtensionHandler* ορίζει τις μεθόδους με τις οποίες ο μηχανισμός αποθηκεύει τα επεκτάσιμα στοιχεία και χαρακτηριστικά (extensibility elements & attributes) ενός εγγράφου WSDL. Η κλάση *SoapAddressParser* υλοποιεί έναν extension handler για ένα συγκεκριμένο στοιχείο του πρωτοκόλλου SOAP. Τέλος, η διεπαφή *IParser* περιέχει τη βασική μέθοδο της εφαρμογής που χρησιμοποιείται προκειμένου να διαβαστεί (parse) ένα έγγραφο WSDL και να αποθηκευτεί στην βάση δεδομένων της cache. Υλοποιείται από την κλάση *Parser* η οποία χρησιμοποιεί τις δυο προηγούμενες διεπαφές.



Σχήμα 6 Βασικά συστατικά του μηχανισμού

Οι άλλες κλάσεις που φαίνονται στο προηγούμενο σχήμα είναι: η *CacheLogger* η οποία αρχικοποιεί τον μηχανισμό ιστορικού (logging) της εφαρμογής, η *CacheService* που περιέχει μια δομή περιγραφής των υπηρεσιών ιστού και η *PropertyFile* για την επεξεργασία του αρχείου *propertyFile* που περιέχει παραμέτρους σχετικά με την εκτέλεση της εφαρμογής.

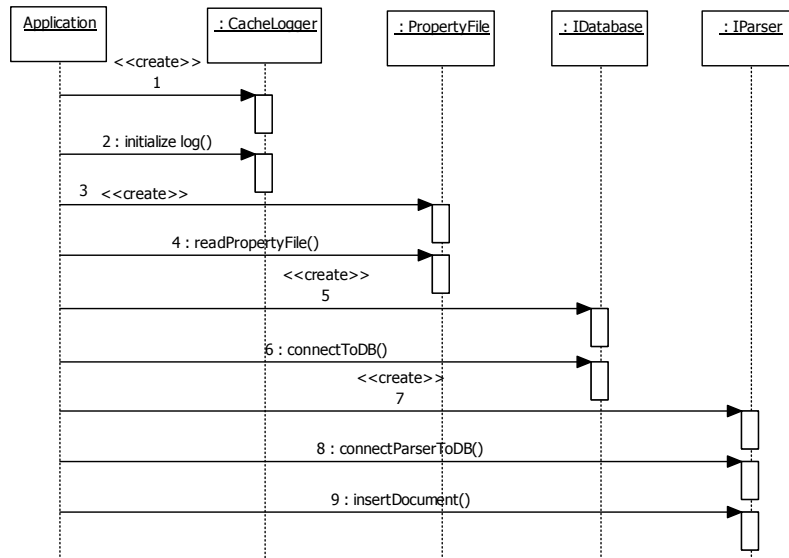
Η κλάση *IDatabase* δημιουργεί ένα αντικείμενο, το οποίο χειρίζεται τη βάση δεδομένων. Το αντικείμενο αυτό συνδέεται με έναν *IParser*. Ο *IParser* αναλαμβάνει να κάνει parse το αρχείο WSDL και να καλέσει ένα αντικείμενο *IExtensionHandler* προκειμένου να διαβάσει στοιχεία επεκτασιμότητας. Τέλος, ο μηχανισμός δεν περιλαμβάνει έλεγχο εισαγωγής του ίδιου αρχείου WSDL από διαφορετικά URL. Ο έλεγχος αυτός πρέπει να υλοποιηθεί στο παραπάνω επίπεδο για την αποφυγή διπλοεγγραφών στη βάση.

6 Υλοποίηση Μηχανισμού Αποθήκευσης

Ο μηχανισμός υλοποιήθηκε με την αντικειμενοστραφή γλώσσα προγραμματισμού Java και συνοδεύεται από ένα γραφικό περιβάλλον (GUI) που υποστηρίζει *εισαγωγή WSDL αρχείων* στη βάση, *διαγραφή WSDL αρχείων* από τη βάση με κλειδί είτε τον προμηθευτή είτε το URL που προσδιορίζει το αρχείο WSDL, *καθαρισμό της βάσης* από τύπους δεδομένων και προμηθευτές στους οποίους δεν υπάρχει καμία αναφορά και τέλος *αναζήτηση των λειτουργιών* (operations) που ικανοποιούν συγκεκριμένα κριτήρια. Παρακάτω περιγράφονται δύο βασικά θέματα υλοποίησης.

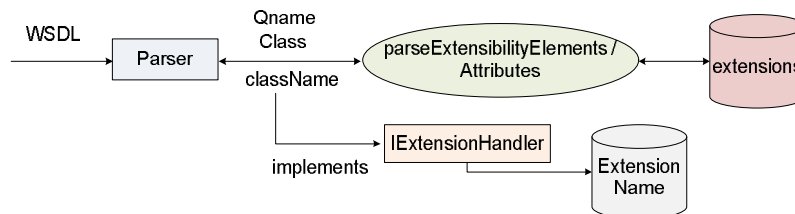
6.1 Ανάγνωση και Αποθήκευση αρχείων WSDL με τον μηχανισμό cache

Στο παρακάτω διάγραμμα (Σχήμα 7) φαίνεται η χρήση του API που υλοποιήθηκε, προκειμένου να γίνει η ανάγνωση και αποθήκευση ενός αρχείου WSDL στην cache.



Σχήμα 7 Διαδικασία ανάγνωσης WSDL αρχείων από τον μηχανισμό αποθήκευσης

6.2 Χειρισμός Επεκτάσεων – Extension Handling



Σχήμα 8 Μηχανισμός χειρισμού επεκτάσεων

Η λειτουργία του χειρισμού επεκτάσεων της γλώσσας WSDL σε επίπεδο εφαρμογής φαίνεται στο παρακάτω σχήμα (Σχήμα 8). Το αντικείμενο *Parser* καλεί την εσωτερική (private) μέθοδο, η οποία ελέγχει αν υπάρχει κλάση που να κάνει parsing το συγκεκριμένο στοιχείο επέκτασης. Ο έλεγχος γίνεται στη βάση δεδομένων και συγκεκριμένα στον πίνακα *extensions*. Η κλάση που επιστρέφεται υλοποιεί τη διεπαφή *IExtensionHandler*. Το αντικείμενο *Parser* ορίζει ένα νέο αντικείμενο της κλάσης που επέστρεψε η *parseExtensibilityElements* και καλεί τη μέθοδο *parse* του *IExtensionHandler* ώστε οι σχετικές πληροφορίες να αποθηκευτούν στον πίνακα με όνομα το όνομα του extension αυτού. Ο πίνακας αυτός μπορεί να ανήκει στο ίδιο σχήμα που ανήκει και ο πίνακας *extensions* ή στο σχήμα κάποιας άλλης βάσης

δεδομένων. Ο μηχανισμός χειρισμού επεκτάσεων είναι ελαστικός και δεν περιορίζει το χρήστη.

7 Συμπεράσματα και Μελλοντικές Κατευθύνσεις

Ο μηχανισμός που υλοποιήθηκε στα πλαίσια της εργασίας αυτής αποθηκεύει «αφαιρετικά» τις πληροφορίες που υπάρχουν στο abstract μέρος μιας WSDL περιγραφής ώστε να βελτιώνεται σημαντικά η διαδικασία αναζήτησης υπηρεσιών ιστού. Η επιλεκτική αποθήκευση των δεδομένων επιτρέπει τη συρρίκνωση του όγκου τους, διευκολύνοντας έτσι την επεξεργασία τους σε τοπικό επίπεδο. Ο τρόπος με τον οποίο έχει υλοποιηθεί ο μηχανισμός χειρισμού επεκτάσεων της WSDL προσφέρει στον χρήστη ευελιξία ώστε να μπορεί να τον διαμορφώνει σύμφωνα με τις ανάγκες του.

Οι μελλοντικές κατευθύνσεις της εργασίας αποτελούνται από οδηγίες σχετικές με τη συντήρηση και βελτίωση του λογισμικού, καθώς και από περαιτέρω πρακτικές για βελτίωση της απόδοσης της μηχανής αναζήτησης. Ανάμεσα στις κατευθύνσεις που προτείνονται είναι η πραγματοποίηση μετρήσεων σχετικά με την κλιμάκωση και απόδοση της εφαρμογής. Για τις μετρήσεις μπορεί να χρησιμοποιηθεί ένας αρκετά μεγάλος αριθμός αρχείων WSDL και το κατάλληλο περιβάλλον που θα προσφέρει υποστήριξη για τις μετρήσεις αυτές. Επίσης, προτείνεται η περαιτέρω εκλέπτυνση του μηχανισμού υποστήριξης επεκτάσεων, έτσι ώστε να διευκολύνεται η επέκταση της εφαρμογής από τρίτους, και στη συνέχεια η ανάπτυξη επεκτάσεων και plugins για την υποστήριξη επεκτάσεων SAWSDL [18]. Ακόμα, υπάρχει η ανάγκη περαιτέρω επεξεργασίας των δεδομένων που αποθηκεύονται στην cache, με σκοπό τη δημιουργία καταλόγων που θα διευκολύνουν τη διαδικασία της αναζήτησης, καθώς και την εφαρμογή τεχνικών Ανάκτησης Πληροφοριών (Information Retrieval) για την εύρεση σχέσεων ομοιότητας (similarity relations) μεταξύ των αποθηκευμένων δεδομένων. Για παράδειγμα, οι σχέσεις αυτές θα μπορούσαν να αφορούν υπηρεσίες που παρουσιάζουν παρόμοια λειτουργικότητα. Τέλος, απαιτείται η ενσωμάτωση και δοκιμή του μηχανισμού από τη μηχανή αναζήτησης υπηρεσιών ιστού USQL Engine.

8 Ευχαριστίες

Βασικός παράγοντας για την περάτωση της εργασίας αυτής, υπήρξε η στενή συνεργασία μου με τον Μιχάλη Πανταζόγλου, υποψήφιο διδάκτορα ΕΚΠΑ, ο οποίος μέσα από τις συναντήσεις που πραγματοποιήθηκαν με βοήθησε να γνωρίσω με κέφι και ενθουσιασμό το χώρο των υπηρεσιών ιστού. Τέλος, θα ήθελα να ευχαριστήσω την καθηγήτριά μου Αφροδίτη Τσαλγατίδου, για τη συνεργασία μας, τις συμβουλές της και τη βοήθεια που μου πρόσφερε σε όλους τους τομείς.

Αναφορές

- [1] Erik Christensen, Francisco Curbera, et al., “Web Services Description Language (WSDL) 1.1”, W3C Note, 15 March 2001.
- [2] David Booth, Hugo Haas, et al., “Web Services Architecture”, W3C Working Group Note, 11 February 2004; <http://www.w3.org/TR/ws-arch/>.
- [3] Ethan Cerami, *Web Services Essentials*, O’ Reilly, February 2002.
- [4] Steve Graham, Simeon Simeonov, et al., *Building Web Services with Java™: Making Sense of XML, SOAP, WSDL, and UDDI*, Sams Publishing, 12 December 2001.
- [5] Ιωάννης Δελβερούδης, *Σχεδιασμός και Υλοποίηση Μηχανής Αναζήτησης Συνδυασμένων Υπηρεσιών για το Web*, Διπλωματική Διατριβή, Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, ΕΜΠ, Οκτώβριος 2003.
- [6] Chris Armstrong, *Modeling Web Services with UML*, 2002; http://www.omg.org/news/meetings/workshops/presentations/WebServices_2002/03-2_Armstrong-ModelingWebServices_with_UML.pdf.
- [7] V. de Castro, E. Macros and B. Vela, Representing WSDL with Extended UML; http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r51_art1_c.pdf.
- [8] Hugo Haas, Allen Brown, “Web Services Glossary”, W3C Working Group Note, 11 February 2004; <http://www.w3.org/TR/ws-gloss/>.
- [9] *Αρχιτεκτονική και δομικά στοιχεία των Web Services*, Πανεπιστήμιο Μακεδονίας; <http://www.it.uom.gr/project/soap/Theory/architecture.html>.
- [10] Aphrodite Tsalgatidou, Thomi Pilioura, *An overview of Standards and Related Technology in Web Services*, Kluwe Academic Publishers, 2002.
- [11] Martin Fowler, “Εισαγωγή στη UML”, Κλειδάριθμος, 3^η αμερικάνικη έκδοση, 2004.
- [12] R. Elmasari, S.B. Navathe, “Θεμελιώδεις Αρχές Συστημάτων Βάσεων Δεδομένων», Τόμος Α’, 4^η έκδοση αναθεωρημένη, εκδόσεις Διάυλος.
- [13] George Feinberg, *Anatomy of a Native XML Database*, Sleepycat Software; <http://www.gca.org/xmlusa/2004/slides/feinberg/Anatomy%20of%20a%20Native%20XML%20Database.ppt>.
- [14] Ed Ort, *Service – Oriented Architecture and Web Services: Concepts, Technologies, and Tools*, Sun Developer Network (SDN), Απρίλιος 2005.
- [15] Graham Glass, *The Web services (r)evolution: Part 1*, IBM, 1^η Νοεμβρίου 2000.
- [16] Pantazoglou, M., Tsalgatidou, A., and Athanasopoulos, G.: Discovering Web Services and JXTA Peer-to-Peer Services in a Unified Manner, In *Proceedings of the 4th International Conference on Service Oriented Computing*, ICSOC 2006, 2006, Chicago, USA.
- [17] Don Box, David Ehnebuske, et al., ”Simple Object Access Protocol (SOAP) 1.1”, W3C Note, 8 May 2000; <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>.
- [18] Joel Farrell, Holger Lausen, “Semantic Annotations for WSDL and XML Schema”, W3C, 27 June 2007; <http://www.w3.org/2002/ws/sawSDL/spec/>.

CPML: Μία Γλώσσα Υψηλού Επιπέδου για Προγραμματισμό με Περιορισμούς

Παύλος Ε. Σουλδάτος
Σταύρος Γ. Πολίτης

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών,
Τμήμα Πληροφορικής και Τηλεπικοινωνιών

Περίληψη Μελετούμε και αναλύουμε τον χώρο του προγραμματισμού με περιορισμούς και παρουσιάζουμε μία δηλωτική γλώσσα υψηλού επιπέδου για τη μοντελοποίηση και επίλυση προβλημάτων στο χώρο αυτό. Σκοπός της, να υποστηρίξει χρήστες μη εξοικειωμένους με γλώσσες προγραμματισμού, όσο και το να λειτουργήσει ως πλατφόρμα για γρήγορη και εύκολη δοκιμή εναλλακτικών μοντέλων για δοθέντα προβλήματα.

Λέξεις κλειδιά: περιορισμός, περιορισμένη μεταβλητή, πεδίο τιμών, αναζήτηση, μοντελοποίηση

1 Εισαγωγή

Ο προγραμματισμός με περιορισμούς είναι μια προκύπτουσα τεχνολογία λογισμικού για τη δηλωτική περιγραφή και αποτελεσματική επίλυση μεγάλων, ιδιαίτερα συνδυαστικών, προβλημάτων ειδικά στους τομείς του μαθηματικού προγραμματισμού, σχεδιασμού ενεργειών και του χρονοπρογραμματισμού. Αντιπροσωπεύει τις πιο συναρπαστικές εξελίξεις στις γλώσσες προγραμματισμού των τελευταίων δύο δεκαετιών και, όπως ήταν αναμενόμενο, έχει αναγνωριστεί από την ACM (Association for Computing Machinery) ως μία από τις στρατηγικές κατευθύνσεις στην έρευνα των υπολογιστών. Όχι μόνο είναι βασισμένος σε ισχυρές θεωρητικές αρχές, αλλά και προσελκύει το εμπορικό ενδιαφέρον, ειδικότερα για τους τομείς της μοντελοποίησης, της βελτιστοποίησης και των προβλημάτων ικανοποίησης περιορισμών.

Η μελέτη μας πάνω στον προγραμματισμό με περιορισμούς ξεκινά με μια εκτενή περιγραφή των εξελίξεων στον χώρο από τη δεκαετία του '60 έως και σήμερα. Μελετώνται οι τεχνικές που έχει αναπτύξει η τεχνολογία λογισμικού έτσι ώστε να λύνει προβλήματα με μεγάλες διαστάσεις δεδομένων. Στη συνέχεια προτείνουμε μία γλώσσα υψηλού επιπέδου για προγραμματισμό με περιορισμούς. Τέλος, παρουσιάζουμε τα βήματα της εξέλιξης αυτής της γλώσσας.

2 Προγραμματισμός με περιορισμούς

Τα τελευταία χρόνια, ο προγραμματισμός με περιορισμούς (constraint

¹ Επιβλέποντες: Σταματόπουλος Παναγιώτης και Ζερβουδάκης Κυριάκος

programming) έχει προσελκύσει την προσοχή επιστημόνων από πολλές περιοχές λόγω της δυνατότητάς του για την επίλυση δύσκολων πραγματικών προβλημάτων. Παρ' όλα αυτά, ο προγραμματισμός με περιορισμούς είναι ακόμα μία από τις λιγότερο γνωστές και κατανοητές τεχνολογίες.

Ο προγραμματισμός με περιορισμούς είναι η μελέτη των υπολογιστικών συστημάτων βασισμένων στους περιορισμούς που τα διέπουν. Αντιπροσωπεύει ένα συνδυασμό των παλαιών εννοιών της ανάλυσης και των δομών συνεπαγωγής των μαθηματικών με διάφορες σύγχρονες τάσεις: γενικές γλώσσες για υπολογιστική αναπαράσταση, αποδοτικότητα της ανάλυσης και της υλοποίησης, ανοχή σε χρήσιμους (αν και ελλιπείς) αλγορίθμους (που συνδέονται ίσως με πιο "αδύνατες" μεθόδους, όπως η αναζήτηση). Όλα αυτά βρίσκονται στην υπηρεσία του σχεδιασμού και της υλοποίησης των συστημάτων προγραμματισμού και επίλυσης προβλημάτων από διαφορετικές περιοχές. Η αρχή σε αυτήν την περιοχή μπορεί να προσδιοριστεί από την έρευνα στην τεχνητή νοημοσύνη (artificial intelligence) και τα γραφικά υπολογιστών (computer graphics) στη δεκαετία του '60 και τη δεκαετία του '70 που εστίασαν στη ρητή αναπαράσταση και χειρισμό περιορισμών στα υπολογιστικά συστήματα. Μόνο στην δεκαετία του '90, εντούτοις, προέκυψε η αυξανόμενη συνειδητοποίηση ότι αυτές οι ιδέες παρέχουν τη βάση για μια ισχυρή προσέγγιση στον προγραμματισμό, τη μοντελοποίηση και την επίλυση προβλημάτων και ότι διαφορετικές προσπάθειες να χρησιμοποιηθούν αυτές οι ιδέες μπορούν να ενοποιηθούν κάτω από ένα κοινό εννοιολογικό και πρακτικό πλαίσιο.

Η βασική ουσία αυτού του πλαισίου είναι ο χωρισμός των ερευνών σε δύο επίπεδα. Το πρώτο επίπεδο είναι αυτό των πολύ γενικά ορισμένων συστημάτων περιορισμών (constraint systems), που είναι συστήματα που παράγουν συμπεράσματα χρησιμοποιώντας τα κομμάτια των μερικών πληροφοριών βασισμένα σε θεμελιώδεις διαδικασίες όπως τη διάδοση περιορισμών (constraint propagation), τη συνέπεια ή συνεπαγωγή (consistency), την ικανοποίηση (satisfaction), την κανονικοποίηση (normalization) και τη βελτιστοποίηση (optimization). Εκτός από παραδοσιακά συστήματα περιορισμών που έχουν ερευνηθεί ήδη κατά τη διάρκεια των αιώνων, όπως προβλήματα που επιλύθηκαν από μαθηματικούς περασμένων αιώνων πριν τον ορισμό του επιστημονικού πεδίου και προβλήματα ακέραιου προγραμματισμού με περιορισμούς, ο προγραμματισμός με περιορισμούς εστιάζει σε μία ευρεία ποικιλία συστημάτων από τα "μη δομημένα" πεπερασμένα πεδία τιμών ("unstructured" finite domains) σε εξισώσεις πάνω σε δέντρα ("term-unification") ως τα χρονικά διαστήματα (temporal intervals). Αυξανόμενη προσοχή δίνεται στην ανακάλυψη αποδοτικών τεχνικών για αυτές τις διαδικασίες περιορισμών απέναντι στις ευρείες κλάσεις τέτοιων συστημάτων περιορισμών, για την ανακάλυψη μίας κοινής εκμεταλλεύσιμης δομής για τα συστήματα περιορισμών.

Το δεύτερο επίπεδο αφορά τις γλώσσες προγραμματισμού που επιτρέπουν στο χρήστη να διευκρινίσει περισσότερες πληροφορίες για το ποιοι περιορισμοί πρέπει να παραχθούν, πώς πρέπει να συνδυαστούν και να επεξεργαστούν κ.λπ. Ίσως μοναδικές στον προγραμματισμό με περιορισμούς είναι οι γλώσσες μοντελοποίησης (modelling languages) που εξερευνούν βασισμένες σε λογικά κατασκευάσματα ελέγχου, όπως ο λογικός προγραμματισμός με περιορισμούς (constraint logic programming) και ο σύγχρονος προγραμματισμός με περιορισμούς (concurrent constraint programming). Αυτές οι γλώσσες αλληλεπιδρούν με το πρώτο επίπεδο καθαρά μέσω των βασικών διαδικασιών περιορισμών. Αυτό παρέχει στο χρήστη ένα πολύ εκφραστικό πλαίσιο (παραμετρικό στο ελλοχεύον σύστημα περιορισμών) για την παραγωγή, το χειρισμό και τον έλεγχο περιορισμών (στην περίπτωση των γλωσσών βασισμένων σε λογική) διατηρώντας το δηλωτικό τους χαρακτήρα. Αυτή η πραγματοποίηση των ενοποιημένων πλαισίων συνοδεύτηκε ταυτόχρονα από την υλοποίηση μερικών

γενικών συστημάτων, τα οποία βρίσκουν ευρεία χρήση σε εφαρμογές από τη μοντελοποίηση φυσικών συστημάτων και ελέγχου ρομπότ ως το χρονοπρογραμματισμό (scheduling) των αποθηκευτικών χώρων των πλοίων στα λιμάνια.

3 Η γλώσσα CPML

Το αρτικόλεξο CPML προέρχεται από τις λέξεις Constraint Programming Modelling Language, δηλαδή γλώσσα μοντελοποίησης προγραμματισμού με περιορισμούς. Είναι μία γλώσσα υψηλού επιπέδου επίλυσης προβλημάτων μαθηματικού προγραμματισμού, που μοντελοποιούνται στα πλαίσια του προγραμματισμού με περιορισμούς. Ένα πρόβλημα ικανοποίησης περιορισμών αντιμετωπίζεται ως ένα σύνολο περιορισμών πάνω στις μεταβλητές του προβλήματος. Η προσπάθεια εύρεσης λύσεων οργανώνεται σε ένα δέντρο λύσεων, το δέντρο αναζήτησης, χρησιμοποιώντας μια μέθοδο ή στρατηγική αναζήτησης.

Η CPML απευθύνεται σε άτομα που δεν είναι αναγκαίο να γνωρίζουν προγραμματισμό ή αρχιτεκτονική υπολογιστών και γενικά ιδιαίτερες τεχνικές λεπτομέρειες, αλλά σε άτομα με γνώσεις πάνω σε προβλήματα μαθηματικού προγραμματισμού και ειδικότερα προγραμματισμού με περιορισμούς, που στοχεύουν να αφιερώσουν χρόνο στη μοντελοποίηση προβλημάτων και εύρεση αποδοτικών μεθόδων αναζήτησης. Γι' αυτό και οι "άχαρες" λεπτομέρειες του προγραμματισμού αποκρύπτονται από τη CPML και ο χρήστης λειτουργεί στο υψηλό επίπεδο αφαίρεσης της γλώσσας.

Η γλώσσα διαθέτει πλούσιο συντακτικό όσον αφορά τις δηλώσεις σταθερών και μεταβλητών που περιγράφουν και ποσοτικοποιούν τις οντότητες ενός φυσικού προβλήματος και όσον αφορά τον ορισμό των περιορισμών και της μεθόδου αναζήτησης. Είναι μία προσπάθεια ενοποίησης των γλωσσών μοντελοποίησης και των γλωσσών προγραμματισμού με περιορισμούς. Από τις πρώτες υιοθετούνται οι υψηλού επιπέδου συμβολισμοί χειρισμού των συνόλων και των πράξεων άλγεβρας. Από τις δεύτερες δανείζεται το πλούσιο υπόβαθρό τους στον ορισμό περιορισμών, την υποστήριξη διαφόρων προβλημάτων, καθολικών περιορισμών και καθορισμού στρατηγικών και διαδικασιών αναζήτησης λύσεων.

Η CPML είναι βασισμένη στον KCSP solver, ο οποίος έχει υλοποιηθεί από τον ερευνητή του τμήματος Πληροφορικής και Τηλεπικοινωνιών της σχολής Θετικών Επιστημών του Εθνικού και Καποδιστριακού Πανεπιστημίου Αθηνών κ.Κυριάκο Ζερβουδάκη. Ο επιλυτής προβλημάτων ικανοποίησης περιορισμών KCSP είναι υλοποιημένος στην αντικειμενοστραφή γλώσσα C++. Ο ενδιάμεσος κώδικας της CPML είναι κώδικας C++, που χρησιμοποιεί τα προσφερόμενα από τον επιλυτή εργαλεία για να επιλύσει ένα πρόβλημα.

Ένα πρόγραμμα της CPML έχει γενικά την εξής δομή:

```
Δηλώσεις;
solve { περιορισμοί };
minimize μεταβλητή
subject to { περιορισμοί };
search { μέθοδος αναζήτησης / στρατηγική };
print { μεταβλητές }
```

Αναλυτικότερα, η τυπική δομή ενός προγράμματος CPML έχει ως εξής:

1. Λίστα δηλώσεων. Περιλαμβάνει όλες τις δηλώσεις. Δηλώσεις μπορούν να γίνουν στα εξής:

A. Δήλωση πεδίων τιμών (Domain Declaration)
 π.χ. *range* my_range 1..3;
 B. Δήλωση απαριθμητών τύπων
 π.χ. *enum* persons {first, second, third};
 Γ. Δήλωση σταθερών (Constant Declaration)
 π.χ. *int* my_constant=10; *int* my_constant_matrix[my_range]=[1,2,3];
 Δ. Δήλωση μεταβλητών (Variable Declaration)
 π.χ. *var int* my_variable; *var int* my_variable_matrix[my_range, 1..8];
 E. Δήλωση περιορισμών (Constraint Declaration)
 π.χ. *constraint* my_constraint my_variable < 5;
 2. Ορισμός (καθολικών) περιορισμών
 π.χ. *solve* { *alldifferent*(my_variable_matrix); my_constraint };
 3.1. Ποσότητα προς βελτιστοποίηση
 π.χ. *minimize* my_var
 3.2. Ορισμός περιορισμών
 π.χ. *subject to* { my_var = *sum*(i in my_range) {my_variable_matrix[i]} }
 4. Μέθοδος αναζήτησης
 π.χ. *search* { *DFSearch*(my_variable_matrix) };
 5. Μεταβλητές προς εκτύπωση
 π.χ. *print* { cost; totalsum }

Συχνά σε προγράμματα περιλαμβάνονται και τα έξι παραπάνω μέρη. Εντούτοις, υπάρχει περίπτωση κάποια τμήματα να παραλειφθούν ή να είναι κενά. Παραδείγματος χάριν μπορεί να μην υπάρχει δήλωση πεδίου τιμών (1A) ή δήλωση περιορισμών (1Δ). Ο χώρος του *solve* (2) μπορεί να παραλειφθεί εντελώς, όπως επίσης μπορεί να παραλειφθεί το 3. Τέλος, ο χώρος του *search* μπορεί να είναι κενός, οπότε, σε αυτή την περίπτωση ακολουθείται μία προεπιλεγμένη μέθοδος αναζήτησης από τη CPML. Συνολικά ένα κενό πρόγραμμα είναι αποδεκτό, χωρίς βέβαια να παράγεται ενδιάμεσος κώδικας.

Η λογική ενός προγράμματος είναι στο πρώτο μέρος του να ορίζονται οι ποσότητες και οντότητες του φυσικού προβλήματος (1) και οι περιορισμοί που υπάρχουν σε αυτό (2,3). Έτσι, δημιουργείται ο χώρος αναζήτησης του προβλήματος. Στη συνέχεια, στο δεύτερο μέρος του προγράμματος, κατά την αναζήτηση (4), ορίζεται ο μέθοδος αναζήτησης πιθανών λύσεων του προβλήματος και, συγκεκριμένα, η μορφή του δέντρου αναζήτησης. Τέλος, το αποτέλεσμα (αν βρεθούν λύσεις) είναι κάποιες εκτυπώσεις τιμών των περιορισμένων μεταβλητών(5).

Η CPML διαθέτει απλές σταθερές και μεταβλητές και παραμέτρους, όπως και πίνακες σταθερών και μεταβλητών. Οι διαθέσιμοι τύποι των σταθερών είναι οι ακέραιοι (*integer*) και οι δυαδικοί (*boolean*), των μεταβλητών είναι οι ακέραιοι (*integer*), οι δυαδικοί (*boolean*) και οι τύποι που παράγονται από το χρήστη. Τέλος, των παραμέτρων είναι οι ακέραιοι (*integer*). Η CPML υποστηρίζει πίνακες μέχρι διάστασης 3.

Οι μεταβλητές που χρησιμοποιούνται στη CPML είναι οι περιορισμένες μεταβλητές (*constrained variables*). Με τον όρο περιορισμένη μεταβλητή εννοούμε ότι μία τέτοια μεταβλητή δηλώνεται έτσι ώστε να μπορεί να λάβει τιμές μέσα από ένα πεδίο τιμών που ορίζεται ρητά κατά τη δήλωσή της. Σχεδόν πάντα οι περιορισμένες μεταβλητές αφορούν ποσότητες του φυσικού προβλήματος και συμμετέχουν σε περιορισμούς που τίθενται.

Επιπλέον των σταθερών και των μεταβλητών, υπάρχουν οι παράμετροι. Οι παράμετροι ορίζονται τοπικά για κάποιο συγκεκριμένο σκοπό. Ουσιαστικά πρόκειται για μεταβλητές μη περιορισμένες, που λαμβάνουν τιμές με τρόπο προκαθορισμένο ή

όχι, από ένα πεδίο τιμών. Δηλώνονται μέσα σε περιοχές των forall, sum, tryall και select, που θα αναφερθούν παρακάτω.

Η CPML διαθέτει μία σειρά τελεστών, όπως ο τελεστής ακολουθίας εντολών (το ελληνικό ερωτηματικό (;)), που δηλώνει σύζευξη εντολών και δομών της γλώσσας, ο τελεστής ανάθεσης τιμής (το ίσον (=)), που χρησιμοποιείται είτε για αρχικοποίηση σταθερών είτε για δημιουργία περιορισμών. Έπειτα, υπάρχουν οι αριθμητικοί τελεστές, που χρησιμοποιούνται για τον ορισμό περιορισμών, οι οποίοι περιλαμβάνουν αριθμητικές πράξεις ανάμεσα σε σταθερές, μεταβλητές και παραμέτρους. Στη συνέχεια, είναι οι τελεστές σύγκρισης, που χρησιμοποιούνται για τον ορισμό περιορισμών και διάταξη παραμέτρων και οι τελεστές ελέγχου τιμών, που είναι οι OnValue, OnRange και OnDomain, που χρησιμοποιούνται κατά την αναζήτηση. Τέλος, υπάρχουν οι λογικοί τελεστές, που χρησιμοποιούνται για να ορίσουν λογικές σχέσεις περιορισμών ή δυαδικών εκφράσεων με λογικές σχέσεις.

Η CPML διαθέτει κάποιες συναρτήσεις, που επιτελούν ορισμένες σημαντικές λειτουργίες ιδιαίτερα χρήσιμες κατά τη φάση αναζήτησης λύσεων ενός προβλήματος. Επομένως, χρησιμοποιούνται μόνο κατά την αναζήτηση. Οι συναρτήσεις που προσφέρονται είναι η dsize, που δέχεται ως όρισμά της μία έκφραση και επιστρέφει το μέγεθος του πεδίου τιμών της έκφρασης, οι συναρτήσεις dmin, dmax και dmid, που δέχονται μία έκφραση και επιστρέφουν το ελάχιστο, το μέγιστο και το μεσαίο στοιχείο του πεδίου τιμών της έκφρασης αντίστοιχα. Επίσης, η συνάρτηση regretdmin δέχεται μία έκφραση και επιστρέφει τη διαφορά μεταξύ των δύο μικρότερων στοιχείων του πεδίου τιμών της έκφρασης. Αν το πεδίο τιμών περιέχει μόνο μία τιμή, τότε επιστρέφει μηδέν. Η συνάρτηση regretdmax συντάσσεται και λειτουργεί με τον ίδιο τρόπο, ενώ επιστρέφει τη διαφορά των δύο μεγαλύτερων τιμών της έκφρασης που δέχεται. Οι συναρτήσεις regretdmin και regretdmax συντάσσονται ακριβώς όπως και η dsize.

Όλες οι παραπάνω συναρτήσεις συμμετέχουν στις εκφράσεις κατά τη φάση αναζήτησης λύσεων. Η παρακάτω συνάρτηση είναι η μοναδική που μπορεί να χρησιμοποιηθεί σε οποιαδήποτε έκφραση, δηλαδή και κατά τον ορισμό του προβλήματος. Η συνάρτηση abs είναι η κλασσική μαθηματική συνάρτηση της απόλυτης τιμής. Δέχεται μία οποιαδήποτε έκφραση της CPML. Σε περίπτωση που η έκφραση είναι σταθερά, τότε επιστρέφεται η απόλυτη τιμή της σταθεράς. Αν είναι μεταβλητή με ένα αντίστοιχο πεδίο τιμών, τότε η abs εκφράζει έναν περιορισμό που έχει το αντίστοιχο "απόλυτο" πεδίο τιμών, που αναλογεί στην έκφραση. Η abs συντάσσεται ακριβώς όπως η dsize.

Οι εκφράσεις στη CPML είναι τριών ειδών. Πρώτον, είναι οι εκφράσεις, που χρησιμοποιούνται κατά τον ορισμό ενός προβλήματος και των περιορισμών του, δεύτερον οι εκφράσεις που χρησιμοποιούνται κατά την αναζήτηση των λύσεων του προβλήματος και τέλος οι δυαδικές εκφράσεις. Οι εκφράσεις ορισμού περιορισμών προβλημάτων είναι αυτές που μπορούν να χρησιμοποιηθούν για να ορίσουν τους περιορισμούς ενός προβλήματος, δηλαδή στο χώρο του solve ή του subject to. Οι εκφράσεις ορισμού αναζήτησης λύσεων προβλημάτων είναι αυτές που μπορούν να χρησιμοποιηθούν για να ορίσουν τη μέθοδο αναζήτησης λύσης ή λύσεων ενός προβλήματος, δηλαδή στο χώρο του search. Οι δυαδικές εκφράσεις μπορούν να χρησιμοποιηθούν και αυτές κατά τη φάση αναζήτησης λύσεων ενός προβλήματος. Η βασική διαφορά τους με τις παραπάνω εκφράσεις είναι ότι ενώ οι ανωτέρω εκφράσεις αντιπροσωπεύουν τιμές αριθμητικές, οι δυαδικές εκφράσεις αντιπροσωπεύουν τιμές δυαδικές, δηλαδή true ως τιμή αληθείας και false ως τιμή ψεύδους.

Οι περιορισμοί αποτελούν ένα σημαντικό κομμάτι του προγραμματισμού με περιορισμούς. Η CPML διαθέτει αρκετές δομές που βοηθούν τον ορισμό περιορισμών και σε συνδυασμό με τις σταθερές, τις μεταβλητές, τους πίνακες, τους

τελεστές και τις συναρτήσεις αποτελεί ένα ισχυρό εργαλείο για τον ορισμό και αναζήτηση λύσεων προβλημάτων περιορισμών.

Οι περιορισμοί στη CPML μπορούν να δηλωθούν σε τρία σημεία ενός προγράμματος. Ένας περιορισμός είναι δυνατό να οριστεί στις δηλώσεις (αρχή προγράμματος), μέσα στο χώρο (block) του solve ή στο χώρο (block) του subject to. Όταν έχουμε δήλωση ενός περιορισμού στην αρχή του προγράμματος, μαζί με τις δηλώσεις σταθερών και μεταβλητών, τότε θεωρείται ότι απλά δηλώνεται και υπάρχει μια μεταβλητή ως ένα όνομα περιορισμού. Για να ισχύσει αυτός ο περιορισμός και να γίνει διάδοση αυτού στις περιορισμένες μεταβλητές που επηρεάζει, πρέπει να δηλωθεί το όνομά του μέσα στο χώρο του solve ή μέσα στο χώρο του subject to. Αυτό υποστηρίζεται για την περίπτωση περιορισμών υπό συνθήκη και για τη δημιουργία ονομάτων και την ευκολία διαχείρισης περιορισμών κατά την κρίση του χρήστη. Περιορισμοί που δηλώνονται μέσα στο χώρο του solve και του subject to ισχύουν αυτόματα και διαδίδονται πριν αρχίσει η επίλυση του προβλήματος. Η διαφορά των δηλώσεων στους χώρους των solve και subject to είναι μικρή. Στο χώρο του solve ενδείκνυται η δήλωση των καθολικών περιορισμών που αφορούν τις μεταβλητές του προβλήματος ως οντότητες και τις τιμές που παίρνουν, χωρίς να απαγορεύεται να γραφεί και ένας άλλος οποιοσδήποτε περιορισμός του φυσικού προβλήματος. Στο χώρο του subject to τίθενται οι γενικοί "μαθηματικοί" περιορισμοί του προβλήματος, σε συσχέτιση με κάποια περιορισμένη μεταβλητή, δηλαδή οι περιορισμοί που συνδέουν τις μεταβλητές εισόδου με την μεταβλητή που μετράει την «ποιότητα» της λύσης. Όταν δεν υπάρχει μία μεταβλητή προς βελτιστοποίηση, δε χρησιμοποιείται ο χώρος του subject to, αλλά μόνο του solve.

Οι περιορισμοί που τίθενται κατά τον ορισμό του προβλήματος περιορίζουν τα πεδία τιμών των μεταβλητών του προβλήματος. Η τομή όλων των περιορισμών τίθεται σε ισχύ, ώστε να δημιουργηθεί ο χώρος λύσεων του προβλήματος.

Οι περιορισμοί της CPML είναι οι καθολικοί, που ορίζουν σχέσεις ανάμεσα σε μονοδιάστατους πίνακες μεταβλητών και οι περιορισμοί με χρήση τελεστών σύγκρισης. Χρήσιμες δομές της γλώσσας αποτελούν οι περιορισμοί με χρήση του "forall", όπου ορίζονται παράμετροι, οι οποίες λαμβάνουν τιμές μέσα από ένα συγκεκριμένο πεδίο τιμών και χρησιμοποιούνται για την απαρίθμηση μεταβλητών πινάκων. Μία άλλη χρήσιμη δομή της γλώσσας για ορισμό περιορισμών είναι η δομή επιλογής "if-then-else".

Η μέθοδος αναζήτησης λύσεων (βέλτιστων ή μη) στο δέντρο αναζήτησης ενός προβλήματος αποτελεί μία στρατηγική απόφασης ως προς τη σειρά με την οποία θα επισκεφτεί το πρόγραμμα τους κόμβους του δέντρου. Γι' αυτό το λόγο και η μέθοδος αναζήτησης είναι ένας σημαντικότερος παράγοντας στον εύκολο και γρήγορο εντοπισμό λύσεων. Η κατάλληλη στρατηγική μπορεί να διαφοροποιήσει το χρόνο εύρεσης βέλτιστων ή καθόλου λύσεων.

Εκτός των γενικών μεθόδων αναζήτησης, που θα περιγραφούν παρακάτω, ο χρήστης μπορεί να ορίσει με χρήση των επιλογών αναζήτησης μία σειρά παραμέτρων που σχετίζονται με τη μέθοδο αναζήτησης. Αυτό έχει ως συνέπεια τη διαφοροποίηση της αναζήτησης για κάθε πρόβλημα. Συγκεκριμένα, η αναζήτηση μπορεί να ορίζει σε ποιο πίνακα θα γίνει μία αναζήτηση, με ποια σειρά οι μεταβλητές του πίνακα θα πάρουν τιμές ώστε να γίνει αναζήτηση λύσεων, με ποια σειρά θα δοθούν οι πιθανές τιμές στις μεταβλητές του πίνακα. Τα κριτήρια που χρησιμοποιούνται είναι καθορισμένα είτε με τρόπο εντελώς αυθαίρετο είτε με κριτήρια διαισθητικά δικαιολογημένα. Με τα παραπάνω μπορεί να οριστεί το δέντρο αναζήτησης ενός προβλήματος.

Η CPML διαθέτει μια σειρά δομών και εκφράσεων που μπορούν να χρησιμοποιηθούν για τον καθορισμό της μεθόδου αναζήτησης ενός προβλήματος.

Στην περίπτωση που δεν έχει οριστεί μέθοδος αναζήτησης από το χρήστη, ακολουθείται μία προκαθορισμένη πρώτα κατά βάθος αναζήτηση. Η CPML με τις εκφράσεις αναζήτησης που διαθέτει δίνει τη δυνατότητα ορισμού διαφορετικών ενεργειών σε διαφορετικές καταστάσεις κατά την αναζήτηση. Παρακάτω παρουσιάζονται πρώτα οι συνθήκες που μπορούν να χρησιμοποιηθούν και στη συνέχεια όλες οι επιλογές αναζήτησης που διατίθενται. Επίσης, η αναζήτηση, σε αντίθεση με το δηλωτικό χαρακτήρα των περιορισμών κατά τον ορισμό του προβλήματος, έχει διαδικαστικό χαρακτήρα, μπορεί δηλαδή να θεωρηθεί ως μία ρουτίνα αναζήτησης λύσεων χωρίς δηλωτική σημασία. Εφ' όσον δεν υπάρχει δηλωτική σημασία, η σειρά των επιλογών έχει καθοριστική σημασία. Με άλλα λόγια, η αναζήτηση εκτελείται σειριακά, όπως ένα διαδικαστικό πρόγραμμα.

Η αναζήτηση, όπως ειπώθηκε, αφορά στον καθορισμό του δέντρου αναζήτησης. Οι διαδικασίες αναζήτησης δεν είναι απαραίτητο να αναθέτουν τιμές σε όλες τις μεταβλητές του προβλήματος. Σε αυτές τις περιπτώσεις, η CPML εφαρμόζει προεπιλεγμένα μία κατά βάθος αναζήτηση για τις μη αποτιμημένες μεταβλητές.

Η CPML υποστηρίζει την κατά βάθος αναζήτηση, αναζήτηση με χρήση του “forall”, αναζήτηση με χρήση του “try / tryall”, αναζήτηση με χρήση της δομής ελέγχου “if-then-else” και αναζήτηση με χρήση των τελεστών ελέγχου τιμών. Επίσης, υπάρχει τοπική αναζήτηση με περιορισμούς, αναζήτηση με χρήση του “select”, αναζήτηση με χρήση του let και αναζήτηση με χρήση της δομής επανάληψης “while”.

4 Παραδείγματα της γλώσσας CPML

Ας θεωρήσουμε το ακέραιο πρόβλημα του σακιδίου (integer knapsack problem). Τότε αυτό το πρόβλημα μπορεί να γραφεί σε CPML ως εξής:

```
/*Knapsack problem*/
int N = ...;
range items 1..N;
int MaxCapacity = ...;
int value[items] = ...;
int weight[items] = ...;
var int take[items] in 0..1;
var int cost in 0..MaxInt;

maximize
cost
subject to
{
    cost = sum(i in items) { value[i] * take[i] };
    sum(i in items) { weight[i] * take[i] } <= MaxCapacity
};
search{DFSsearch(take)};
print {cost; take}
```

Στο ανωτέρω παράδειγμα προσπαθούμε να πετύχουμε το μέγιστο όφελος χωρίς να ξεπεράσουμε την χωρητικότητα του σακιδίου. Εκτελείται πρώτα κατά βάθος αναζήτηση με αποτίμηση των μεταβλητών του πίνακα take.

Το ευρέως γνωστό πρόβλημα των N – Βασιλισσών στην κλασική μορφή του

ζητά να τοποθετηθούν 8 Βασίλισσες σε μία σκακιέρα 8x8 έτσι ώστε να μην απειλούν η μία την άλλη. Η μοντελοποίηση αυτού του προβλήματος αποτελεί ένα αρκετά καλό παράδειγμα επίδειξης των δυνατοτήτων της CPML. Παρακάτω θεωρούμε το πρόβλημα για μεταβλητό μέγεθος σκακιέρας. Το πρόβλημα μοντελοποιείται ως εξής:

```
/*N – Queens problem*/
int N = ...;
range Domain 1..N;
enum Dim {col, row};
var Domain queen[Dim, Domain];
solve
{
  forall (d in Dim & i in Domain & j in Domain: i <> j)
  {
    queen[d, i] <> queen[d, j];
    queen[d, i] + i <> queen[d, j] + j;
    queen[d, i] - i <> queen[d, j] - j
  }
  forall (i in Domain & v in Domain)
  if (queen[col, i] = v) then queen[row, v] = i
  else if queen[row, v] = i then queen[col, i] = v endif
  endif
};

search
{
  forall (i in Domain ordered by increasing < dsize(queen[col, i]) >)
  tryall (v in Domain ordered by increasing < dsize(queen[row, i]) >)
  queen[col, i] = v
  onFailure
  queen[col, i] <> v
};

print {queen}
```

Στο ανωτέρω παράδειγμα προσπαθούμε να βρούμε μία οποιαδήποτε λύση του προβλήματος. Παρατηρούμε ότι η αναζήτηση διαφοροποιείται από την απλή πρώτα κατά βάθος και χρησιμοποιείται η μέθοδος πρώτα ο καλύτερος με κριτήριο το μέγεθος του εναπομείναντος πεδίου τιμών τις γραμμής της μεταβλητής για τη σειρά αποτίμησης μεταβλητών και κριτήριο το μέγεθος του πεδίου τιμών της στήλης της μεταβλητής για τη σειρά επιλογής τιμής για τη μεταβλητή.

5 Υλοποίηση της γλώσσας CPML

Ο σκοπός της υλοποίησης της γλώσσας ήταν η δημιουργία ενός γρήγορου μεταγλωττιστή για τη γλώσσα που να καλύπτει όλες τις σχεδιαστικές προδιαγραφές. Ο μεταγλωττιστής λαμβάνει ένα αρχείο γραμμένο σε CPML και παράγει κώδικα C++ κατάλληλο ώστε να χρησιμοποιεί τον KCSP solver, ώστε να επιλύεται ένα πρόβλημα. Όπως προαναφέραμε, ο χρήστης δε χρειάζεται να γνωρίζει προγραμματισμό σε C++, παρά αρκείται στη μοντελοποίηση του προβλήματος και μόνο.

Για την υλοποίηση του μεταγλωττιστή επιλέξαμε τα μεταεργαλεία παραγωγής λεκτικών και συντακτικών αναλυτών flex και bison, που αποτελούν μετεξελιξείς των lex και yacc, αντίστοιχα. Τα μεταεργαλεία lex και yacc δημιουργήθηκαν τη δεκαετία του 1970 από την AT&T ως συνοδευτικά προγράμματα του λειτουργικού συστήματος Unix. Αποφασίσαμε να υλοποιήσουμε ένα λεκτικό αναλυτή με όλες τις λεκτικές μονάδες της γλώσσας και δύο συντακτικούς αναλυτές, έναν για τον ορισμό μεταβλητών και περιορισμών και έναν για τον ορισμό της μεθόδου αναζήτησης. Επίσης, επιλέχθηκε για την παραγωγή του τελικού κώδικα να γίνεται μόνο ένα πέρασμα του αρχείου εισόδου από το μεταγλωττιστή.

Όλες οι μεταβλητές ενός προγράμματος CPML μετατρέπονται στο ενδιάμεσο επίπεδο(C++), έτσι ώστε να αποφύγουμε συγκρούσεις ονόματος μεταβλητών με λέξεις κλειδιά της C++. Έστω ότι ο χρήστης ορίζει μία μεταβλητή a, τότε στον ενδιάμεσο κώδικα χειριζόμαστε τη μεταβλητή `_a`. Ομοίως, μεταβλητές που ορίζονται εσωτερικά από το μεταγλωττιστή στην ενδιάμεσο κώδικα ακολουθούν τη σύμβαση να μπαίνει ο χαρακτήρας `'_'` (underscore) πριν από αυτές. Έτσι, απαγορεύεται μόνο ο ορισμός λεκτικών μονάδων που είναι δεσμευμένες από τη γλώσσα CPML και είναι ορισμένες στο λεκτικό αναλυτή.

Διάφορες συναρτήσεις, που είναι πιθανώς απαραίτητες κατά την εκτέλεση, υλοποιούνται σε ξεχωριστό αρχείο C++, το οποίο μεταγλωττίζεται με το κυρίως ενδιάμεσο πρόγραμμα. Από αυτές τις συναρτήσεις κάποιες είναι σταθερές για όλα τα προγράμματα, ενώ άλλες δημιουργούνται δυναμικά στο αρχείο, σε όσα προγράμματα απαιτούνται, από τους κανόνες παραγωγής. Επίσης, πολλά ενδιάμεσα αρχεία δημιουργούνται ανά περίπτωση και χρησιμοποιούνται για την παραγωγή του ενδιάμεσου κώδικα. Αυτά ήταν απαραίτητα για να υλοποιείται η παραγωγή ενδιάμεσου κώδικα σε ένα πέρασμα του αρχείου εισόδου.

Ένα σημαντικό κομμάτι ενός μεταγλωττιστή είναι η ανάνηψη από λάθη με χρήση του εργαλείου που προσφέρει το μεταεργαλείο bison. Το μεταεργαλείο bison εκπίπτει σε κατάσταση λάθους όταν συναντήσει συντακτικό λάθος, και μόνο. Εφόσον, τα σημασιολογικά λάθη είναι υπεύθυνος ο προγραμματιστής να τα αντιμετωπίσει. Στο μεταεργαλείο, αυτό τα λάθη αντιμετωπίζονται με συντακτικούς κανόνες που περιέχουν τη δεσμευμένη λέξη error, ακολουθούμενη από μία λέξη κλειδί. Η είσοδος αγνοείται έως ότου συναντήσουμε τη λέξη κλειδί. Έπειτα, συνεχίζεται η μεταγλώττιση του προγράμματος από το τελευταίο σημείο. Η συγκεκριμένη μέθοδος ανάνηψης από λάθη μπορεί να οδηγήσει, λανθασμένα, στην εύρεση επιπλέον σφαλμάτων, ως αποτελέσματα προηγούμενων λαθών.

Μία συνάρτηση γραμμένη σε C αναλαμβάνει να ενώσει όλα τα παραπάνω και να εκτελέσει τη συντακτική ανάλυση του πρώτου μέρους καλώντας τον πρώτο συντακτικό αναλυτή και στη συνέχεια του δεύτερου μέρους καλώντας το δεύτερο. Αναλαμβάνει επίσης το καθάρισμα των αρχείων εισόδου που δημιουργήθηκαν για τις ανάγκες του κάθε προγράμματος.

Ο ενδιάμεσος κώδικας C++ χρησιμοποιεί ορισμένες κλάσεις του KCSP solver. Έτσι, θα παρατηρήσουμε ότι, κατά το πρότυπο του προγραμματισμού με περιορισμούς, υπάρχει μία αποθήκη περιορισμών και ένας μηχανισμός διάδοσης περιορισμών. Για να υποστηριχθούν αυτά, υπάρχει ένας διαχειριστής μνήμης και ένας διαχειριστής προβλήματος.

Σε κάθε πρόβλημα υπάρχει ένα γενικό πλάνο στο αρχείο του ενδιάμεσου κώδικα C++, σύμφωνα με το οποίο δηλώνονται πρώτα οι καθολικές μεταβλητές και έπειτα στη συνάρτηση main ορίζεται το πρόβλημα και καλείται η συνάρτηση αναζήτησης και η συνάρτηση εκτύπωσης αποτελεσμάτων, ο ορισμός των οποίων ακολουθεί τον ορισμό της συνάρτησης main.

6 Αναφορές

1. Β. Ζησιμόπουλος, Αλγοριθμική επιχειρησιακή έρευνα, Πανεπιστημιακές εκδόσεις, Αθήνα, 2005.
2. P. Van Hentenryck, L. Perron and J.-F. Puget, "Search and Strategies in OPL", *ACM Transactions on Computational Logic*, vol. 1, no. 2, Oct. 2000, pp. 285-320.
3. P. Van Hentenryck, *The OPL Optimization Programming Language*, MIT Press, 1999.
4. P. Van Hentenryck, *Constraint Satisfaction in Logic Programming*, MIT Press, 1989.
5. P. Van Hentenryck, L. Michel, L. Perron and J.-C. Regin, "Constraint Programming in OPL", *Proc. Int'l. Conf. on Principles and Practice of Declarative Progr.* (PPDP '99), Paris, France, September 1999.
6. P. Van Hentenryck, V. Saraswat et al, "Strategic Directions in Constraint Programming", *ACM Computing Surveys*, vol. 28, no. 4, Dec. 1996.
7. *A C++ Implementation of CLP*, Ilog Solver Collected papers, Ilog tech report, J.-F. Puget, 1994.
8. E. Tsang, *Foundations of Constraint Satisfaction*, Academic Press, 1993.
9. C. Voudouris, *Guided Local Search for Combinatorial Optimization Problems*, doctoral dissertation, Dept. Computer Science, Univ. Essex, Colchester, UK, 1997.
10. D. S. Reeves and K. G. Shin, "Parallel and Distributed Real-Time Computing", *IEEE Parallel and Distributed Technology*, vol. 2, no. 4, 1994, p. 8.
11. T. Walsh, "Stochastic constraint programming", *Proc. 15th Europ. Conf. in Artificial Intelligence* (ECAI '02), IOS Press, 2002.
12. Z. Kiziltan and T. Walsh, "Constraint Programming with Multisets", *Proc. 8th Int'l Conf. in Constraint Programming* (CP '02), Springer, 2002.
13. R. Bartak, "Constraint Programming: In Pursuit of the Holy Grail", *Proc. Week of Doct. Stud.* (WDS '99), MatFyzPress, Charles University, Prague, 1999.
14. E. C. Freuder, "Synthesizing Constraint Expressions", *Communications ACM*, vol. 21, no. 11, Nov. 1978, pp. 958-966.
15. E. C. Freuder and R. J. Wallace, "Partial Constraint Satisfaction", *Artificial Intelligence ACM*, vol. 58, no. 1-3, 1998, pp. 21-70.
16. E. Best, F. S. De Boer, and C. Palamidessi, "Concurrent constraint programming with information removal", *Proc. 1st Conf. Concurrent Constraint Prog.*, 1995.
17. M. Codish, M. Falaschi and K. Marriot, "Suspension analysis of concurrent constraint logic programs", *ACM Trans. Programming Language Systems*, vol. 16, no. 3, 1994.
18. R. Dechter, "Enhancement schemes for constraint processing: Backjumping, learning and cutset decomposition", *Artificial Intelligence* vol. 41, no. 3, Jan. 1990, pp. 273-312.
19. E. C. Freuder, "Principles and Practice of Constraint Programming", *Proc. 2nd Int'l Conf. in Constraint Prog.* (CP '96). No. 1118 in LNCS, Springer-Verlag, 1996.
20. H. Hobose, S. Matsuoka, A. Yonezawa, "Generalized local propagation: A framework for solving constraint hierarchies", *Proc. Int'l Conf. Principles and Practice of Constraint Prog.*, vol. 1118, Springer-Verlag, Boston, 1996.

Σχεδίαση με VHDL και Υλοποίηση σε FPGA του Χαμηλότερου Ιεραρχικά Επιπέδου του Συστήματος Μνήμης Cache Level 1, με Επαλήθευση Ορθής Λειτουργίας, στον Υψηλής Απόδοσης Επεξεργαστή “Αθηνά” Αρχιτεκτονικής RISC

Ιωάννης Σταμούλιας std03229@di.uoa.gr
Κωνσταντίνος Μαούνης std03182@di.uoa.gr

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών
Τμήμα Πληροφορικής και Τηλεπικοινωνιών
Πανεπιστημιούπολη, Ιλίσια, 15784, Αθήνα, Ελλάς

Περίληψη

Η πτυχιακή αυτή εργασία έχει ως στόχο την υλοποίηση των τριών τοπολογιών κρυφής μνήμης και την ενσωμάτωσή τους στον υψηλής απόδοσης επεξεργαστή “Αθηνά” αρχιτεκτονικής RISC.

Στα πλαίσια της πτυχιακής εργασίας έγινε η σχεδίαση των τριών τοπολογιών κρυφής μνήμης στην γλώσσα περιγραφής υλικού VHDL. Η υλοποίηση των τριών αυτών τοπολογιών έγινε για FPGA με το πρόγραμμα WebPACK Xilinx. Έγινε επαλήθευση ορθής λειτουργίας με χρήση προγράμματος προσομοίωσης (Modelsim MXE).

Τέλος οι κρυφές μνήμες ενσωματώθηκαν στον υψηλής απόδοσης επεξεργαστή “Αθηνά” αρχιτεκτονικής RISC και έγινε επαλήθευση ορθής λειτουργίας με χρήση προγράμματος προσομοίωσης.

Λέξεις Κλειδιά: Σχεδίαση, Υλοποίηση, FPGA, Έλεγχος ορθής λειτουργίας, Κρυφή μνήμη άμεσης απεικόνισης, Συσχετιστική κρυφή μνήμη συνόλου, Πλήρως συσχετιστική κρυφή μνήμη, Αλγόριθμος αντικατάστασης LRU

1 Εισαγωγή

Η κρυφή μνήμη πρώτου επιπέδου είναι ο ταχύτερος τύπος μνήμης μετά τους καταχωρητές που συναντάμε σε έναν υπολογιστή. Όλοι οι σύγχρονοι επεξεργαστές έχουν μια μικρή, υψηλής ταχύτητας κρυφή μνήμη κοντά στον επεξεργαστή για να κρατάει τα πιο πρόσφατα χρησιμοποιημένα δεδομένα και εντολές από τη μνήμη. Η αρχή της τοπικότητας ισχυρίζεται ότι αν ο επεξεργαστής αναφέρθηκε σε μια περιοχή της μνήμης, είναι πολύ πιθανό να αναφερθεί ξανά σε αυτή πολύ σύντομα. Χρησιμοποιώντας μια κρυφή μνήμη για να κρατάει τις πιο πρόσφατα χρησιμοποιημένες τιμές της μνήμης, ο επεξεργαστής δε χρειάζεται κάθε φορά να πηγαίνει σε αυτή, για να τις φορτώσει. Αυτό παρέχει μια σημαντική αύξηση της απόδοσης, καθώς η μνήμη του συστήματος είναι πολύ πιο αργή από την κρυφή μνήμη του επεξεργαστή.

Η κρυφή μνήμη του επεξεργαστή ονομάζεται πρώτου επιπέδου καθώς είναι η πιο κοντινή μνήμη στον επεξεργαστή. Κάθε φορά που ο επεξεργαστής απαιτεί

πληροφορίες από τη μνήμη ο ελεγκτής της κρυφής μνήμης χρησιμοποιεί ειδικά κυκλώματα για να ελέγξει αν τα δεδομένα είναι ήδη στην κρυφή μνήμη. Αν είναι, ο επεξεργαστής εξοικονομεί το χρόνο της πρόσβασης στην κύρια μνήμη. Η κρυφή μνήμη πρώτου επιπέδου είναι ταχύτερη καθώς τρέχει με την ταχύτητα του επεξεργαστή και είναι ενσωματωμένη μέσα σε αυτόν. Οι κρυφές μνήμες χωρίζονται στις τοπολογίες κρυφή μνήμη άμεσης απεικόνισης, συσχετιστική κρυφή μνήμη συνόλου και πλήρως συσχετιστική κρυφή μνήμη.

2 Τοπολογίες Κρυφής Μνήμης

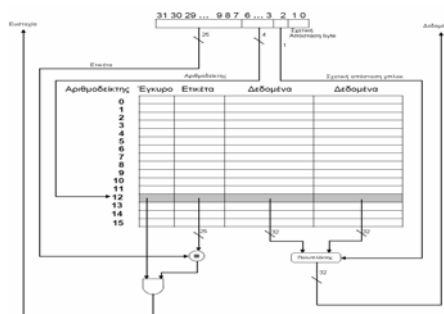
Κρυφή μνήμη άμεσης απεικόνισης

Η κρυφή μνήμη άμεσης απεικόνισης είναι μια δομή κρυφής μνήμης στην οποία κάθε θέση μνήμης απεικονίζεται σε ακριβώς μια θέση στην κρυφή μνήμη. Η τυπική αντιστοιχία διευθύνσεων και θέσεων της κρυφής μνήμης είναι συνήθως απλή. Η απεικόνιση που χρησιμοποιείται είναι:

(Διεύθυνση μπλοκ) modulo (πλήθος μπλοκ κρυφής μνήμης)

Ακολουθεί η εικόνα 1 που φαίνεται πως είναι θεωρητικά μια κρυφή μνήμη άμεσης απεικόνισης. Για την κρυφή μνήμη αυτή έχει υποτεθεί ότι αποτελείται από 16 γραμμές (cache lines), 2 λέξεις ανά μπλοκ και το μέγεθος κάθε λέξης είναι 32 bit.

Η διεύθυνση χωρίζεται στα εξής 4 πεδία: πεδίο σχετικής απόστασης, πεδίο σχετικής απόστασης μπλοκ, πεδίο αριθμοδείκτη και πεδίο ετικέτας.



Εικόνα 1: Κρυφή μνήμη των 180 byte που περιέχει 16 μπλοκ με 2 λέξεις ανά μπλοκ

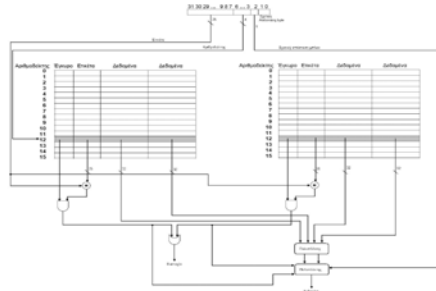
Συσχετιστική κρυφή μνήμη συνόλου

Η συσχετιστική κρυφή μνήμη συνόλου είναι μια δομή κρυφής μνήμης στην οποία κάθε θέση μνήμης μπορεί να απεικονίζεται σε ένα συγκεκριμένο σύνολο από θέσεις της κρυφής μνήμης. Ο αριθμός των θέσεων σε κάθε τέτοιο σύνολο είναι συγκεκριμένος (τουλάχιστον δύο) και ορίζει τον βαθμό συσχέτισης. Η τυπική αντιστοιχία διευθύνσεων και θέσεων της κρυφής μνήμης είναι συνήθως απλή. Η απεικόνιση που χρησιμοποιείται είναι:

(Διεύθυνση μπλοκ) modulo (πλήθος μπλοκ κρυφής μνήμης)

Ακολουθεί η εικόνα 2 που φαίνεται πως είναι θεωρητικά μια συσχετιστική κρυφή μνήμη συνόλου. Για την κρυφή μνήμη αυτή έχει υποτεθεί ότι αποτελείται από 16 γραμμές (cache lines), 2 λέξεις ανά μπλοκ, 2 σύνολα και το μέγεθος κάθε λέξης είναι 32 bit.

Η διεύθυνση χωρίζεται στα εξής 4 πεδία: πεδίο σχετικής απόστασης byte, πεδίο σχετικής απόστασης μπλοκ, πεδίο αριθμοδείκτη, πεδίο ετικέτας.



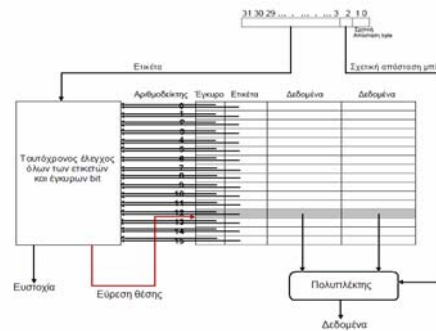
Εικόνα 2: Κρυφή μνήμη των 360 byte που περιέχει 2 σύνολα των 16 μπλοκ με 2 λέξεις ανά μπλοκ

Πλήρως συσχετιστική κρυφή μνήμη

Η πλήρως συσχετιστική κρυφή μνήμη είναι μια δομή κρυφής μνήμης στην οποία κάθε θέση μνήμης απεικονίζεται σε οποιαδήποτε θέση στην κρυφή μνήμη.

Ακολουθεί η εικόνα 3 που φαίνεται πως είναι θεωρητικά μια πλήρως συσχετιστική κρυφή μνήμη. Για την κρυφή μνήμη αυτή έχει υποθεθεί ότι αποτελείται από 16 γραμμές (cache lines), 2 λέξεις ανά μπλοκ και το μέγεθος κάθε λέξης είναι 32 bit.

Η διεύθυνση χωρίζεται στα εξής 3 πεδία: πεδίο σχετικής απόστασης byte, πεδίο σχετικής απόστασης μπλοκ, πεδίο ετικέτας.



Εικόνα 3: Κρυφή μνήμη των 180 byte που περιέχει 16 μπλοκ με 2 λέξεις ανά μπλοκ

3 Σχεδίαση Κρυφών Μνημών

Κρυφή μνήμη άμεσης απεικόνισης

Μετά την υλοποίηση στη γλώσσα περιγραφής υλικού VHDL η κρυφή μνήμη άμεσης απεικόνισης αποτελείται από τα 4 εξής στοιχεία (component) :

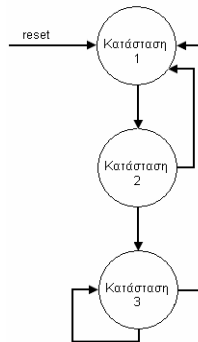
- Στοιχείο που αποθηκεύονται τα δεδομένα (λέξεις).
- Στοιχείο που αποθηκεύονται οι ετικέτες.

- Στοιχείο που αποθηκεύονται τα bit που δηλώνουν αν είναι έγκυρα τα μπλοκ της μνήμης.
- Στοιχείο που συγχρονίζει την λειτουργία της κρυφής μνήμης (FSM).

Για την δημιουργία των στοιχείων που είναι υπεύθυνα για την αποθήκευση των δεδομένων και των ετικετών χρησιμοποιήθηκαν block ram μνήμες. Οι μνήμες αυτές είναι generic ως προς το μέγεθός τους.

Το στοιχείο που είναι υπεύθυνο για την αποθήκευση των bit εγκυρότητας αποτελείται από registers, έτσι ώστε να είναι δυνατή η αρχικοποίησή τους οποιαδήποτε στιγμή της λειτουργίας της κρυφής μνήμης. Το μέγεθος του στοιχείου αυτού είναι generic.

Για την κρυφή μνήμη άμεσης απεικόνισης τα στάδια που μπορεί να περιέλθει είναι αυτά που φαίνονται στην εικόνα 4 που ακολουθεί.



Εικόνα 4: Καταστάσεις της κρυφή μνήμη άμεσης απεικόνισης

Οι ενεργοποίηση της μηχανής καταστάσεων γίνεται με την ύπαρξη αίτησης διαβάσματος ή γραψίματος από τον επεξεργαστή.

Κατάσταση 1: Μηδενίζονται όσα τοπικά σήματα χρησιμοποιηθούν και με το που υπάρξει κάποια αίτηση για διάβασμα ή γράψιμο στέλνεται το σήμα για το stall του επεξεργαστή και πηγαίνει στην κατάσταση 2.

Κατάσταση 2: Στην κατάσταση αυτή ανάλογα του ελέγχου των ετικετών, του bit εγκυρότητας και του τι ενέργεια ζητήθηκε έχουμε τις εξής λειτουργίες:

- Για την περίπτωση που οι ετικέτες είναι ίσες, το μπλοκ είναι έγκυρο και έχει ζητηθεί διάβασμα τότε στέλνουμε τα δεδομένα (λέξη) που βρέθηκαν σταματάμε το stall και επιστρέφουμε στην κατάσταση 1.
- Για την περίπτωση που οι ετικέτες είναι ίσες, το μπλοκ είναι έγκυρο και έχει ζητηθεί γράψιμο τότε στέλνουμε τα δεδομένα από τον επεξεργαστή στις υπόλοιπες μονάδες προς αποθήκευση, στέλνουμε την διεύθυνση και τα δεδομένα και στην κύρια μνήμη (λόγω write through), σταματάμε το stall και επιστρέφουμε στην κατάσταση 1.
- Για την περίπτωση που οι ετικέτες δεν είναι ίσες ή το μπλοκ δεν είναι έγκυρο (ή και τα δύο) και έχει ζητηθεί διάβασμα τότε στέλνει στην κύρια μνήμη την διεύθυνση της πρώτης λέξης για το μπλοκ ώστε να ξεκινήσει η διαδικασία του γεμίσματος του μπλοκ (cache line refill) και πάει στην κατάσταση 3.

- ο Για την περίπτωση που οι ετικέτες δεν είναι ίσες ή το μπλοκ δεν είναι έγκυρο (ή και τα δύο) και έχει ζητηθεί γράψιμο τότε στέλνουμε τα δεδομένα από τον επεξεργαστή στις υπόλοιπες μονάδες προς αποθήκευση, στέλνουμε την διεύθυνση και τα δεδομένα και στην κύρια μνήμη (λόγω write through), σταματάμε το stall και επιστρέφουμε στην κατάσταση 1.

Κατάσταση 3: Στην κατάσταση αυτή γράφονται οι λέξεις που στέλνονται από την κύρια στην κρυφή μνήμη και στην περίπτωση που δεν έχει τελειώσει το cache line refill στέλνεται η επόμενη διεύθυνση στην κύρια μνήμη. Αν έχουμε φτάσει στην τελευταία διεύθυνση τότε αποθηκεύουμε την ετικέτα και το έγκυρο bit, σταματάμε το stall και επιστρέφουμε στην κατάσταση 1 διαφορετικά επαναλαμβάνουμε την κατάσταση 3. Όταν μας σταλθεί το δεδομένο (λέξη) που ζητήθηκε από τον επεξεργαστή του το στέλνουμε.

Συσχετιστική κρυφή μνήμη συνόλου

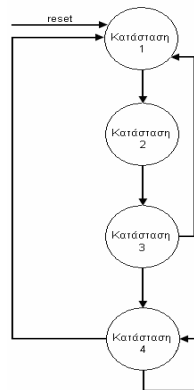
Μετά την υλοποίηση στη γλώσσα περιγραφής υλικού VHDL η συσχετιστικής κρυφής μνήμης συνόλου αποτελείται από τα 4 εξής στοιχεία (component) :

- ο Στοιχείο που αποθηκεύονται τα δεδομένα (λέξεις).
- ο Στοιχείο που αποθηκεύονται οι ετικέτες.
- ο Στοιχείο που αποθηκεύονται τα bit που δηλώνουν αν είναι έγκυρα τα μπλοκ της μνήμης.
- ο Στοιχείο που συγχρονίζει την λειτουργία της κρυφής μνήμης (FSM).

Για την δημιουργία των στοιχείων που είναι υπεύθυνα για την αποθήκευση των δεδομένων και των ετικετών χρησιμοποιήθηκαν block ram μνήμες. Οι μνήμες αυτές είναι generic ως προς το μέγεθός τους.

Το στοιχείο που είναι υπεύθυνο για την αποθήκευση των bit εγκυρότητας αποτελείται από registers, έτσι ώστε να είναι δυνατή η αρχικοποίησή τους οποιαδήποτε στιγμή της λειτουργίας της κρυφής μνήμης. Το μέγεθος του στοιχείου αυτού είναι generic.

Για την συσχετιστική κρυφή μνήμη συνόλου τα στάδια που μπορεί να περιέλθει είναι αυτά που φαίνονται στην εικόνα 5 που ακολουθεί.



Εικόνα 5: Καταστάσεις της συσχετιστικής κρυφής μνήμης συνόλου

Οι ενεργοποίηση της μηχανής καταστάσεων γίνεται με την ύπαρξη αίτησης διαβάσματος ή γραψίματος από τον επεξεργαστή.

Κατάσταση 1: Στην κατάσταση αυτή μηδενίζονται όσα τοπικά σήματα χρησιμοποιηθούν και με το που υπάρξει κάποια αίτηση για διάβασμα ή γράψιμο στέλνεται το σήμα για το stall του επεξεργαστή και πηγαίνει στην κατάσταση 2.

Κατάσταση 2: Στην κατάσταση αυτή γίνεται παράλληλος έλεγχος όλων των ετικετών (από κάθε σύνολο) και όλων των bit εγκυρότητας, δημιουργούνται σήματα για το αν και που βρέθηκε η λέξη και πηγαίνει στην κατάσταση 3.

Κατάσταση 3: Στην κατάσταση αυτή ανάλογα των σημάτων που δημιουργήθηκαν στην κατάσταση 2 και του τι ενέργεια ζητήθηκε έχουμε τις εξής λειτουργίες:

- ο Για την περίπτωση που οι ετικέτες είναι ίσες, το μπλοκ είναι έγκυρο και έχει ζητηθεί διάβασμα τότε στέλνουμε τα δεδομένα (λέξη) που βρέθηκαν σταματάμε το stall και επιστρέφουμε στην κατάσταση 1.
- ο Για την περίπτωση που οι ετικέτες είναι ίσες, το μπλοκ είναι έγκυρο και έχει ζητηθεί γράψιμο τότε στέλνουμε τα δεδομένα από τον επεξεργαστή στις υπόλοιπες μονάδες προς αποθήκευση, στέλνουμε την διεύθυνση και τα δεδομένα και στην κύρια μνήμη (λόγω write through), σταματάμε το stall και επιστρέφουμε στην κατάσταση 1.
- ο Για την περίπτωση που οι ετικέτες δεν είναι ίσες ή το μπλοκ δεν είναι έγκυρο (ή και τα δύο) και έχει ζητηθεί διάβασμα τότε στέλνει στην κύρια μνήμη την διεύθυνση της πρώτης λέξης για το μπλοκ (ώστε να ξεκινήσει η διαδικασία του γεμίματος του μπλοκ (cache line refill) και πάει στην κατάσταση 4.
- ο Για την περίπτωση που οι ετικέτες δεν είναι ίσες ή το μπλοκ δεν είναι έγκυρο (ή και τα δύο) και έχει ζητηθεί γράψιμο τότε στέλνουμε τα δεδομένα από τον επεξεργαστή στις υπόλοιπες μονάδες προς αποθήκευση, στέλνουμε την διεύθυνση και τα δεδομένα και στην κύρια μνήμη (λόγω write through), σταματάμε το stall και επιστρέφουμε στην κατάσταση 1.

Κατάσταση 4: Στην κατάσταση αυτή γράφονται οι λέξεις που στέλνονται από την κύρια στην κρυφή μνήμη και στην περίπτωση που δεν έχει τελειώσει το cache line refill στέλνεται η επόμενη διεύθυνση στην κύρια μνήμη. Αν έχουμε φτάσει στην τελευταία διεύθυνση τότε αποθηκεύουμε την ετικέτα και το έγκυρο bit, σταματάμε το stall και επιστρέφουμε στην κατάσταση 1 διαφορετικά επαναλαμβάνουμε την κατάσταση 4. Όταν μας σταλθεί το δεδομένο (λέξη) που ζητήθηκε από τον επεξεργαστή του το στέλνουμε.

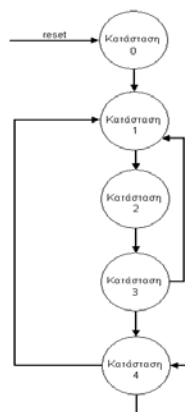
Πλήρως συσχετιστική κρυφή μνήμη

Μετά την υλοποίηση στη γλώσσα περιγραφής υλικού VHDL η πλήρως συσχετιστική κρυφή μνήμη αποτελείται από τα 2 εξής στοιχεία (component) :

- ο Στοιχείο που αποθηκεύονται τα δεδομένα (λέξεις).
- ο Στοιχείο που συγχρονίζει την λειτουργία της κρυφής μνήμης (FSM).

Για την δημιουργία του στοιχείου που είναι υπεύθυνο για την αποθήκευση των δεδομένων χρησιμοποιήθηκε block ram μνήμη. Οι μνήμη αυτή είναι generic ως προς το μέγεθός τους.

Για την πλήρως συσχετιστική κρυφή μνήμη τα στάδια που μπορεί να περιέλθει είναι αυτά που φαίνονται στην εικόνα 6 που ακολουθεί.



Εικόνα 6: Καταστάσεις της πλήρως συσχετιστικής κρυφής μνήμης

Οι ενεργοποίηση της μηχανής καταστάσεων γίνεται με την ύπαρξη αίτησης διαβάσματος ή γραψίματος από τον επεξεργαστή.

Κατάσταση 0: Στην κατάσταση αυτή βρίσκεται μόνο μετά από reset. Αρχικοποιεί όλα τα bit εγκυρότητας, τις ετικέτες και τα σήματα που είναι υπεύθυνα για το address resolving. Τα bit εγκυρότητας και ετικέτες βρίσκονται εσωτερικά στο FSM γιατί πρέπει να ελέγχονται όλες ταυτόχρονα, δεν μπορούν να αναγνωριστούν σαν RAM μνήμες για αυτό το λόγο.

Κατάσταση 1: Στην κατάσταση αυτή μηδενίζονται όσα τοπικά σήματα χρησιμοποιηθούν και με το που υπάρξει κάποια αίτηση για διάβασμα ή γράψιμο στέλνεται το σήμα για το stall του επεξεργαστή και πηγαίνει στην κατάσταση 2.

Κατάσταση 2: Στην κατάσταση αυτή γίνεται παράλληλος έλεγχος όλων των ετικετών και όλων των bit εγκυρότητας, δημιουργούνται σήματα για το αν και που βρέθηκε η λέξη και πηγαίνει στην κατάσταση 3.

Κατάσταση 3: Στην κατάσταση αυτή ανάλογα των σημάτων που δημιουργήθηκαν στην κατάσταση 2 και του τι ενέργεια ζητήθηκε έχουμε τις εξής λειτουργίες:

- Για την περίπτωση που οι ετικέτες είναι ίσες, το μπλοκ είναι έγκυρο και έχει ζητηθεί διάβασμα τότε στέλνουμε τα δεδομένα (λέξη) που βρέθηκαν σταματάμε το stall και επιστρέφουμε στην κατάσταση 1.
- Για την περίπτωση που οι ετικέτες είναι ίσες, το μπλοκ είναι έγκυρο και έχει ζητηθεί γράψιμο τότε στέλνουμε τα δεδομένα από τον επεξεργαστή στις υπόλοιπες μονάδες προς αποθήκευση, στέλνουμε την διεύθυνση και τα δεδομένα και στην κύρια μνήμη (λόγω write through), σταματάμε το stall και επιστρέφουμε στην κατάσταση 1.
- Για την περίπτωση που οι ετικέτες δεν είναι ίσες ή το μπλοκ δεν είναι έγκυρο (ή και τα δύο) και έχει ζητηθεί διάβασμα τότε στέλνει στην κύρια μνήμη την διεύθυνση της πρώτης λέξης για το μπλοκ (ώστε να ξεκινήσει η διαδικασία του γεμίματος του μπλοκ (cache line refill) και πάει στην κατάσταση 4.
- Για την περίπτωση που οι ετικέτες δεν είναι ίσες ή το μπλοκ δεν είναι έγκυρο (ή και τα δύο) και έχει ζητηθεί γράψιμο τότε στέλνουμε τα δεδομένα από τον επεξεργαστή στις υπόλοιπες μονάδες προς αποθήκευση,

στέλνουμε την διεύθυνση και τα δεδομένα και στην κύρια μνήμη (λόγω write through), σταματάμε το stall και επιστρέφουμε στην κατάσταση 1.

Κατάσταση 4: Στην κατάσταση αυτή γράφονται οι λέξεις που στέλνονται από την κύρια στην κρυφή μνήμη και στην περίπτωση που δεν έχει τελειώσει το cache line refill στέλνεται η επόμενη διεύθυνση στην κύρια μνήμη. Αν έχουμε φτάσει στην τελευταία διεύθυνση τότε αποθηκεύουμε την ετικέτα και το έγκυρο bit, σταματάμε το stall και επιστρέφουμε στην κατάσταση 1 διαφορετικά επαναλαμβάνουμε την κατάσταση 4. Όταν μας σταλθεί το δεδομένο (λέξη) που ζητήθηκε από τον επεξεργαστή του το στέλνουμε.

Αλγόριθμοι Αντικατάστασης LRU

Αλγόριθμοι αντικατάστασης χρησιμοποιήθηκαν στις τοπολογίες συσχετιστική κρυφή μνήμη συνόλου και πλήρως συσχετιστική κρυφή μνήμη.

Λόγω της αρχής της χρονικής τοπικότητας, θα έπρεπε να αντικαθίσταται το δεδομένο που είχε αποθηκευτεί πιο παλιά χρονικά στην μνήμη, έτσι υλοποιήθηκε και χρησιμοποιήθηκε ο αλγόριθμος αντικατάστασης LRU. Στην κρυφή μνήμη η LRU χρησιμοποιείται σε κάθε περίπτωση ασχέτως αν ζητείται εγγραφή ή διάβασμα, αν υπάρχει ή όχι το ζητούμενο.

Στην συσχετιστική κρυφή μνήμη συνόλου στις περιπτώσεις που δεν υπάρχει το ζητούμενο η LRU γίνεται για κάθε σύνολο στο μπλοκ που αντιστοιχείται από την διεύθυνση που στέλνει ο επεξεργαστής και στις περιπτώσεις που υπάρχει το ζητούμενο η LRU γίνεται για κάθε σύνολο που βρίσκεται δεξιά του συνόλου που περιέχει την εγγραφή, στο μπλοκ που αντιστοιχείται από την διεύθυνση που στέλνει ο επεξεργαστής.

Στην πλήρως συσχετιστικής κρυφής μνήμης στις περιπτώσεις που δεν υπάρχει το ζητούμενο η LRU γίνεται για κάθε μπλοκ και στις περιπτώσεις που υπάρχει το ζητούμενο η LRU γίνεται για κάθε μπλοκ που βρίσκεται κάτω από το μπλοκ που περιέχει την ζητούμενη λέξη.

4 Αποτελέσματα Σύνθεσης

Για την σύνθεση των τριών τοπολογιών κρυφής μνήμης έχουμε τις εξής ρυθμίσεις: “Target device: xc3s200”, “Package: ft256”.

Τα αποτελέσματα τις σύνθεσης φαίνονται στους ακόλουθους πίνακες:

Η σύνθεση που παρουσιάζεται είναι για την κρυφή μνήμη άμεσης απεικόνισης και αποτελείται από 16 γραμμές με 2 λέξεις ανά γραμμή.

| Logic Utilization | Used | Available | Utilization |
|-----------------------------|------|-----------|-------------|
| Number of Slides | 258 | 1920 | 13% |
| Number of Slides Flip Flops | 113 | 3840 | 2% |
| Number of 4 input LUTs | 477 | 3840 | 12% |
| Number of bonded IOBs | 200 | 173 | 115% |
| Number of GCLKs | 7 | 8 | 87% |

Πίνακας 1: Αποτελέσματα σύνθεσης κρυφής μνήμης άμεσης απεικόνισης

Η σύνθεση που παρουσιάζεται είναι για την συσχετιστική κρυφή μνήμη συνόλου και αποτελείται από 2 σύνολα 16 γραμμών με 2 λέξεις ανά γραμμή.

| Logic Utilization | Used | Available | Utilization |
|-----------------------------|------|-----------|-------------|
| Number of Slides | 433 | 1920 | 22% |
| Number of Slides Flip Flops | 322 | 3840 | 8% |
| Number of 4 input LUTs | 810 | 3840 | 21% |
| Number of bonded IOBs | 200 | 173 | 115% |
| Number of GCLKs | 8 | 8 | 100% |

Πίνακας 2: Αποτελέσματα σύνθεσης συσχετιστικής κρυφής μνήμης συνόλου

Η σύνθεση που παρουσιάζεται είναι για την πλήρως συσχετιστική κρυφή μνήμη και αποτελείται από 8 γραμμών με 2 λέξεις ανά γραμμή.

| Logic Utilization | Used | Available | Utilization |
|-----------------------------|------|-----------|-------------|
| Number of Slides | 457 | 1920 | 23% |
| Number of Slides Flip Flops | 352 | 3840 | 9% |
| Number of 4 input LUTs | 823 | 3840 | 21% |
| Number of bonded IOBs | 200 | 173 | 115% |
| Number of GCLKs | 8 | 8 | 100% |

Πίνακας 3: Αποτελέσματα σύνθεσης πλήρως συσχετιστική κρυφή μνήμη

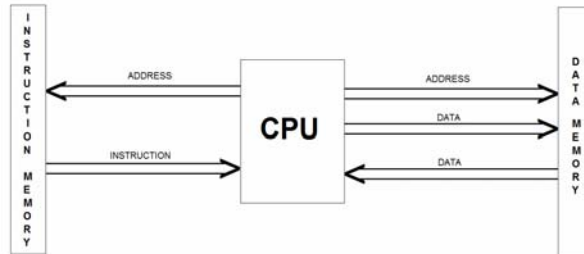
Για σύνθεση σε ίδιο μέγεθος μνημών εκείνη η τοπολογία που καταλαμβάνει τον μικρότερο χώρο είναι η κρυφή μνήμη άμεσης απεικόνισης, ακολουθεί με μικρή διαφορά η συσχετιστική κρυφή μνήμη συνόλου και τέλος η πλήρως συσχετιστική κρυφή μνήμη, η οποία καταλαμβάνει πολύ μεγαλύτερο χώρο. Χαρακτηριστικά φαίνεται και από τα αποτελέσματα σύνθεσης, που δίνονται πιο πάνω, ότι η πλήρως συσχετιστική κρυφή μνήμη καταλαμβάνει δύο φορές σχεδόν το χώρο που καταλαμβάνει η άμεσης απεικόνισης, ενώ σε μέγεθος είναι η μισή από αυτή. Ο λόγος είναι ότι η υλοποίηση των μέσων αποθήκευσης των ετικετών και των bit εγκυρότητας γίνεται με χρήση καταχωρητών στη πλήρως συσχετιστική κρυφή μνήμη.

Μεγαλύτερο ρυθμό ευστοχίας έχεις η πλήρως συσχετιστική κρυφή μνήμη, ακολουθεί η συσχετιστική κρυφή μνήμη συνόλου και τέλος κρυφή μνήμη άμεσης απεικόνισης.

Η τοπολογία κρυφής μνήμης με τα καλύτερα χαρακτηριστικά είναι η *συσχετιστική κρυφή μνήμη συνόλου*, η οποία έχει καλά ποσοστά ευστοχίας και δεν απαιτεί μεγάλο χώρο.

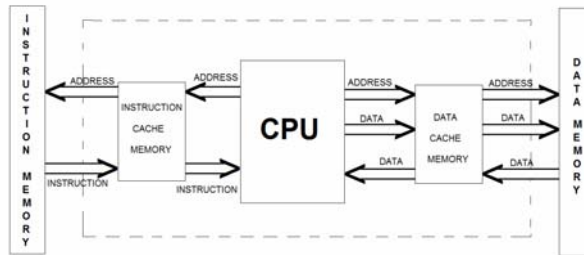
5 Ενσωμάτωση Κρυφών Μνημών στον Επεξεργαστή “Αθηνά”

Ο επεξεργαστής «Αθηνά» πριν την δημιουργία των κρυφών μνημών επικοινωνούσε απευθείας με τις εξωτερικές μνήμες. Οι μνήμες αυτές είναι υλοποιημένες ως αρχεία αποθήκευσης δεδομένων με αποτέλεσμα να μην είναι συνθέσιμες. Μια θεωρητική αναπαράσταση της σύνδεσης αυτής φαίνεται στην εικόνα 7 που ακολουθεί.



Εικόνα 7: Ένωση επεξεργαστή με εξωτερικές μνήμες

Μετά την δημιουργία των κρυφών μνημών ο επεξεργαστής επικοινωνεί με τις εξωτερικές μνήμες μέσω αυτών. Έτσι επιτυγχάνεται η συνθεσιμότητα του επεξεργαστή αφού το μη υλοποιήσιμο κομμάτι βρίσκεται τώρα πια ένα επίπεδο πιο πάνω. Η σύνδεση του υπολογιστή με τις κρυφές μνήμες και τις εξωτερικές μνήμες φαίνεται στην εικόνα 8 που ακολουθεί.



Εικόνα 8: Ένωση επεξεργαστή με κρυφές και εξωτερικές μνήμες

Κατά την ενσωμάτωση των κρυφών μνημών χρειάστηκε να γίνουν κάποιες αλλαγές στον τρόπο λειτουργίας τους έτσι, ώστε να συγχρονίζονται με τον επεξεργαστή και μεταξύ τους. Οι κύριες αλλαγές έγιναν στη μονάδα ελέγχου (FSM) και αφορούν τον τρόπο που ξεκινούν και εναλλάσσονται οι καταστάσεις. Οι αλλαγές είναι ανεξάρτητες από τον τρόπο οργάνωσης της κρυφής μνήμης και είναι οι εξής:

Αρχικά αρχικοποιείται ένα σήμα που δηλώνει αν έχει έρθει νέα αίτηση γραψίματος ή διαβάσματος. Ακολουθεί έλεγχος αν πρόκειται για μνήμη εντολών ή δεδομένων.

Στην περίπτωση που λειτουργεί ως μνήμη εντολών εναλλάσσονται οι καταστάσεις της μονάδας ελέγχου χωρίς κανένα περιορισμό.

Στην περίπτωση που λειτουργεί ως μνήμη δεδομένων εναλλάσσονται οι καταστάσεις της μονάδας ελέγχου χωρίς κανένα περιορισμό αλλά όταν πρόκειται για νέα αρχή των καταστάσεων θα πρέπει να υπάρχει κάποιο σήμα γραψίματος ή διαβάσματος και να μην έχει εξυπηρετηθεί η ίδια αίτηση νωρίτερα.

Συμπεράσματα

Με την δημιουργία των κρυφών μνημών και την ενσωμάτωσή τους στον υψηλής απόδοσης επεξεργαστή “Αθηνά”, το μη συνθέσιμο κομμάτι που υπήρχε μεταφέρθηκε εκτός του επεξεργαστή. Αυτό είχε ως αποτέλεσμα τη συνθεσιμότητα του επεξεργαστή “Αθηνά”, αρχιτεκτονικής RISC.

Αναφορές

1. David A. Patterson, John L. Hennessy, “Computer Organization and Design The Hardware/Software Interface/3rd Edition”
2. Linda Null, Julia Lobur, “The essentials of Computer Organization and Architecture”
3. John L. Hennessy, David A. Patterson “Computer Architecture - A Quantitative Approach”
4. F. Vahid, T. Givargis, J. Wiley & Sons, “Embedded Systems Design, a unified hardware/software introduction”
5. Ben Cohen, “VHDL Coding Styles & Methodologies”
6. William L. Lynch, “The interaction of Virtual Memory and Cache Memory”
7. Αντώνης Πασχάλης, “Σχεδίαση Ψηφιακών Συστημάτων”, Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών
8. Μαρία Γρηγοριάδου, “Αρχιτεκτονική Γ”, Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών
9. Peter J. Ashenden, “The VHDL Cookbook”