



ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**Λογικός προγραμματισμός με προτιμήσεις
στην απειρότιμη λογική**

Αντώνιος Κ. Τρουμπούκης

Επιβλέπων: Παναγιώτης Ροντογιάννης, Αναπληρωτής Καθηγητής ΕΚΠΑ

ΑΘΗΝΑ

ΑΠΡΙΛΙΟΣ 2012

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Λογικός προγραμματισμός με προτιμήσεις
στην απειρότιμη λογική

Αντώνιος Κ. Τρουμπούκης

A.M.: M1050

ΕΠΙΒΛΕΠΩΝ:

Παναγιώτης Ροντογιάννης, Αναπληρωτής Καθηγητής ΕΚΠΑ

ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ:

Παναγιώτης Ροντογιάννης, Αναπληρωτής Καθηγητής ΕΚΠΑ

Παναγιώτης Σταματόπουλος, Επίκουρος Καθηγητής ΕΚΠΑ

Περίληψη

Στην παρούσα διπλωματική εργασία προτείνουμε μια νέα επέκταση του λογικού προγραμματισμού. Αρχικά, εισάγουμε δύο τελεστές προτίμησης με τη σημασία «προαιρετικά» και «εναλλακτικά» αντίστοιχα, που μπορούν να εμφανίζονται στα σώματα των κανόνων και στη συνέχεια ορίζουμε σημασιολογία σταθερού σημείου για τη νέα αυτή γλώσσα. Η σημασιολογία βασίζεται στην απειρότιμη λογική, μια επέκταση της κλασικής δίτιμης λογικής στην οποία χρησιμοποιούνται άπειρες τιμές αλήθειας «λιγότερο αληθείς» από την τιμή T και άπειρες τιμές αλήθειας «λιγότερο ψευδείς» από την τιμή F . Η διαβάθμιση αυτή στις τιμές αληθείας αντιστοιχεί στον βαθμό προτίμησης.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Γλώσσες Προγραμματισμού

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Λογικός Προγραμματισμός, Σημασιολογία Σταθερού Σημείου, Προτιμήσεις, Απειρότιμη Λογική, Μη-κλασικές Λογικές

Abstract

In this thesis we propose a new extension of Logic Programming. We introduce two unary preference operators with the meaning “optionally” and “alternatively” respectively, which can occur in the bodies of the clauses. In addition, we define fixed-point semantics for this new language. The semantics is based on Infinitesimal Logic, an extension of classical two-valued logic, in which there are infinite truth values that are “less true” than “standard” truth, and infinite truth values that are “less false” than “standard” falsity. These different levels of truth values correspond to different degrees of preferences.

SUBJECT AREA: Programming Languages

KEYWORDS: Logic Programming, Fixed-point Semantics, Preferences, Infinitesimal Logic, Non-classical Logics

Ευχαριστίες

Η παρούσα διπλωματική εργασία πραγματοποιήθηκε στο πλαίσιο του Προγράμματος Μεταπτυχιακών Σπουδών του Τμήματος Πληροφορικής και Τηλεπικοινωνιών του Εθνικού και Καποδιστριακού Πανεπιστημίου Αθηνών. Φτάνοντας στο τέλος αυτής της προσπάθειας, νιώθω την ανάγκη να κοιτάξω λίγο πίσω στο χρόνο και να ευχαριστήσω όλους αυτούς που με άμεσο ή έμμεσο τρόπο συνέβαλαν στην ολοκλήρωση της.

Πρώτα από όλους, θέλω να ευχαριστήσω τον επιβλέποντα μου, Π. Ροντογιάννη. Οι συναρπαστικές διαλέξεις του στο μάθημα μου μετέδωσαν την αγάπη για την Θεωρητική Πληροφορική ενώ οι συζητήσεις μας στο γραφείο του το «μικρόβιο» της έρευνας. Τον ευχαριστώ επίσης για την εμπιστοσύνη που μου έδειξε και για την καθοδήγησή του καθ' όλη τη διάρκεια της εκπόνησης της εργασίας αυτής.

Ευχαριστώ τους καθηγητές μου στο πρόγραμμα μεταπτυχιακών σπουδών του τμήματος για τις γνώσεις που μου προσέφεραν. Δεν μπορώ να μην ξεχωρίσω τους Β. Ζησιμόπουλο, Α. Κιαγιά και ιδιαίτερα τον Σ. Κολλιόπουλο, καθέννας από τους οποίους μου έδειξε ένα διαφορετικό τρόπο να αντιμετωπίζω την έρευνα. Επίσης, ευχαριστώ τον Π. Σταματόπουλο που δέχτηκε να συμμετάσχει στην εξεταστική επιτροπή της διπλωματικής μου.

Τέλος, ευχαριστώ την οικογένειά μου. Τους γονείς μου, Κώστα και Τασία, και την αδερφή μου, Λένα. Η υπομονή, η στήρηξη και η ενθάρρυνσή τους αποτέλεσαν τον σημαντικότερο παράγοντα για να μπορέσω να φτάσω μέχρι εδώ.

Περιεχόμενα

Περίληψη	5
Ευχαριστίες	9
Πρόλογος	15
1 Στοιχεία θεωρίας συνόλων	19
1.1 Σύνολα	19
1.2 Διάταξη συνόλων	20
1.3 Φράγματα και πλήρη πλέγματα	21
1.4 Σταθερά σημεία	22
2 Λογικός προγραμματισμός	27
2.1 Προγραμματίζοντας με λογική	27
2.2 Συντακτικό	28
2.3 Μοντελοθεωρητική σημασιολογία	30
2.4 Σημασιολογία σταθερού σημείου	33
2.5 Επεκτάσεις του λογικού προγραμματισμού	35
3 Απειρότιμη λογική και προτιμήσεις	37
3.1 Απειρότιμη λογική	37

3.2	Προτιμήσεις στην απειρότιμη λογική	39
3.3	Τελεστές προτίμησης	42
3.4	Λογικές προτάσεις με προτιμήσεις	44
4	Λογικός προγραμματισμός με προτιμήσεις	49
4.1	Συντακτικό	49
4.2	Μοντελοθεωρητική σημασιολογία	51
4.3	Σημασιολογία σταθερού σημείου	52
4.3.1	Μερική διάταξη	53
4.3.2	Τελεστής T_P	54
4.3.3	Μονοτονικότητα του T_P	54
4.3.4	Συνέχεια του T_P	56
4.4	Τομή μοντέλων	61
5	Εφαρμογές	65
5.1	Προτιμήσεις σε ξενοδοχεία	65
5.2	Προτιμήσεις στο οδικό δίκτυο	68
6	Σχετικές προσεγγίσεις στη βιβλιογραφία	71
6.1	Λογική με προτιμήσεις	71
6.2	Λογικός προγραμματισμός με προτιμήσεις	72
6.2.1	Προτιμήσεις για προβλήματα βελτιστοποίησης	72
6.2.2	Προτιμήσεις για μη μονοτονικότητα	75
6.3	Απειρότιμη λογική	76
7	Παρατηρήσεις και μελλοντικές κατευθύνσεις	79
7.1	Λειτουργική σημασιολογία	79

7.2	Στρωματοποίηση λογικών προγραμμάτων με προτιμήσεις	80
7.3	Επεκτάσεις της γλώσσας	82
7.4	Προς μια απειρότιμη θεωρία συνόλων	84
Γλωσσάρι		87
	Ελληνο-αγγλικό	87
	Αγγλο-ελληνικό	88
Βιβλιογραφία		91

Πρόλογος

Κάθε πελάτης μπορεί να έχει αυτοκίνητο σε ό,τι χρώμα προτιμά, αρκεί να είναι μαύρο.

Henry Ford, 1863-1947

Κανονίστε να πάμε διακοπές σε όποιο νησί θέλετε, θα προτιμούσα όμως να μην πάμε με πλοίο γιατί ζαλίζομαι.

Ένας φίλος

Ας θεωρήσουμε το σενάριο στο οποίο μια παρέα ατόμων κανονίζουν να πάνε μαζί διακοπές. Αυτός που τελικά θα κληθεί να τις οργανώσει (συνήθως ο πιο υπομονετικός!), θα πρέπει να ακολουθήσει μια διαδικασία στην οποία λαμβάνοντας υπόψη τις απαιτήσεις και τις προτιμήσεις των ατόμων, να καταλήξει σε ένα ή περισσότερα αποτελέσματα. Πολύ πιθανόν (και μάλλον σίγουρα όσο μεγαλύτερη γίνεται η παρέα) ο τελικός προορισμός των διακοπών να μην είναι η πρώτη επιλογή για κάποια άτομα.

Σε τέτοιου είδους εφαρμογές αποσκοπεί ο λογικός προγραμματισμός με προτιμήσεις, η νέα γλώσσα που προτείνουμε σε αυτήν την εργασία. Η γλώσσα αυτή, είναι μια επέκταση του κλασικού λογικού προγραμματισμού, που έχει όμως τα εξής δύο επιπλέον χαρακτηριστικά:

- Περιέχει δύο τελεστές προτίμησης με τη σημασία «προαιρετικά» και «εναλλακτικά» αντίστοιχα, οι οποίοι μπορούν να εμφανίζονται στα σώματα των κανόνων.
- Οι απαντήσεις που παίρνει ο χρήστης δεν είναι απλά *True* ή *False*, αλλά μπορούν να είναι τιμές «λιγότερο αληθείς» από την τιμή *True* ή «λιγότερο ψευδείς» από την τιμή *False*.

Ας εξηγήσουμε το παραπάνω διαισθητικά με δύο παραδείγματα. Η ερώτηση στον ταξιδιωτικό πράκτορα «θέλω ένα αεροπορικό εισιτήριο για Αθήνα και προαιρετικά θέση στο παράθυρο» θα μπορούσε να έχει τρεις απαντήσεις: «ναί», «όχι, δεν έχω

θέση στο παράθυρο» και «όχι, δεν έχω εισητήριο για Αθήνα». Τα δύο «όχι» όμως δεν έχουν την ίδια βαρύτητα, αφού ο προορισμός είναι υποχρεωτική επιλογή ενώ η θέση στο παράθυρο είναι προαιρετική επιλογή, οπότε μπορούμε να πούμε ότι η δεύτερη απάντηση είναι «λιγότερο ψευδής» από την τρίτη. Έτσι έχουμε εισάγει μια τιμή αληθείας «λιγότερο ψευδή» από την τιμή *False*.

Συμμετρικά, η ερώτηση στον ταξιδιωτικό πράκτορα «θέλω να πάω στην Θεσσαλονίκη με τρένο ή εναλλακτικά με λεωφορείο» θα μπορούσε να έχει τρεις απαντήσεις: «ναί, με τρένο», «ναι, με λεωφορείο» και «όχι». Τα δύο «ναι» όμως δεν έχουν την ίδια βαρύτητα, αφού το τρένο είναι πρωτεύουσα επιλογή ενώ το λεωφορείο εναλλακτική, οπότε μπορούμε να πούμε ότι η δεύτερη απάντηση είναι «λιγότερο αληθής» από την πρώτη. Έτσι έχουμε εισάγει μια τιμή αληθείας «λιγότερο αληθή» από την τιμή *True*.

Παρατηρούμε ότι προσθέσαμε ένα επίπεδο αληθοτιμών λιγότερο σημαντικών από τις «ακραίες» *True*, *False*. Γενικεύοντας, μπορούμε να προσθέσουμε δύο, τρία ή και παραπάνω επίπεδα. Προσθέτοντας άπειρα, φτάνουμε να ορίσουμε το πεδίο της απειρότιμης λογικής, μιας επέκτασης της κλασικής δίτιμης λογικής στην οποία χρησιμοποιούνται άπειρες τιμές αλήθειας «λιγότερο αληθείς» από την τιμή *True* και άπειρες τιμές αλήθειας «λιγότερο ψευδείς» από την τιμή *False*. Η απειρότιμη λογική ορίστηκε αρχικά από τους Ροντογιάννη, Wadge στο [42] με σκοπό τον ορισμό μοντελοθεωρητικής σημασιολογίας για τον λογικό προγραμματισμό με άρνηση.

Πως όμως μπορούμε να χρησιμοποιήσουμε τους τελεστές «προαιρετικά» και «εναλλακτικά» (οι οποίοι συμβολίζονται με μ και ω αντίστοιχα) στη νέα αυτή γλώσσα; Ας θεωρήσουμε ότι θέλουμε να κλείσουμε ένα δίκλινο δωμάτιο για διακοπές. Η τιμή θέλουμε να είναι 40€ ή εναλλακτικά 50€, ενώ θα προτιμούσαμε να διαθέτει κλιματισμό. Ένα λογικό πρόγραμμα που θα αντιστοιχεί στις προτιμήσεις αυτές είναι το παρακάτω:

$$\text{rent}(X) \leftarrow \text{afford}(X) \wedge \text{beds}(X, 2) \wedge \mu \text{ac}(X).$$

$$\text{afford}(X) \leftarrow \text{cost}(X, 40\text{€}) \vee \omega \text{cost}(X, 50\text{€}).$$

Ας εξετάσουμε τις τιμές που παίρνει καθένα από τα κατηγορήματα του παραπάνω προγράμματος. Για ένα δωμάτιο a το $\text{afford}(a)$ παίρνει την τιμή

- T_0 , αν το κόστος διαμονής είναι 40€,
- T_1 , αν το κόστος διαμονής είναι 50€,
- F_0 , σε κάθε άλλη περίπτωση.

ενώ το $\text{rent}(a)$ παίρνει την τιμή

- F_0 , αν το δωμάτιο δεν είναι δίκλινο,

- F_1 , αν δεν διαθέτει κλιματισμό,
- που παίρνει το $\text{afford}(a)$, σε κάθε άλλη περίπτωση.

Συνεπώς, ανάλογα με την τιμή του $\text{rent}(a)$ μπορούμε να αποφασίσουμε αν θα κλείσουμε ή όχι το δωμάτιο a . Δηλαδή, αν το $\text{rent}(a)$ έχει πάρει τιμή

- T_0 , τότε θα το κλείσουμε γιατί πληρούνται όλες οι προϋποθέσεις: είναι δίκλινο, διαθέτει κλιματισμό και το κόστος διαμονής είναι χαμηλό.
- T_1 , τότε μάλλον θα το κλείσουμε, αν και το κόστος διαμονής είναι 50€, το οποίο είναι εναλλακτική επιλογή για το κόστος διαμονής.
- F_1 , τότε μάλλον δεν θα το κλείσουμε γιατί λείπει ο κλιματισμός, ο οποίος όμως είναι προαιρετική επιλογή και δεν ενοχλεί τόσο η απουσία του.
- F_0 , τότε δεν θα το κλείσουμε γιατί δεν πληρείται κάποια από τις απαραίτητες προϋποθέσεις: μπορεί να μην είναι δίκλινο ή το κόστος διαμονής να μην είναι ούτε 40€ ούτε 50€.

Μετά από τη σύντομη αυτή εισαγωγή, ας παρουσιάσουμε τη δομή του υπόλοιπου κειμένου της εργασίας: Στο πρώτο κεφάλαιο θα παρουσιάσουμε κάποια στοιχεία από τη θεωρία συνόλων που θα χρειαστούν στα επόμενα κεφάλαια και στο δεύτερο κεφάλαιο θα κάνουμε μια σύντομη αναφορά στον κλασικό λογικό προγραμματισμό, παρουσιάζοντας το συντακτικό και τη σημασιολογία του. Στο τρίτο κεφάλαιο θα περιγράψουμε την απειρότιμη λογική και θα εισάγουμε τους τελεστές προτίμησης, με τη σημασία «προαιρετικά» και «εναλλακτικά». Στο τέταρτο κεφάλαιο θα ορίσουμε τον λογικό προγραμματισμό με προτιμήσεις χρησιμοποιώντας τους τελεστές προτίμησης στα σώματα των κανόνων, και θα προτείνουμε σημασιολογία για τη νέα αυτή γλώσσα. Στο πέμπτο κεφάλαιο θα παρουσιάσουμε κάποιες εφαρμογές του λογικού προγραμματισμού με προτιμήσεις και στο έκτο κεφάλαιο θα εξετάσουμε τις εναλλακτικές προσεγγίσεις που υπάρχουν στη βιβλιογραφία σε σχέση με τη δική μας δουλειά. Τέλος, στο έβδομο και τελευταίο κεφάλαιο θα συνοψίσουμε παραθέτοντας κάποιες παρατηρήσεις, καθώς και πιθανές μελλοντικές κατευθύνσεις για την έρευνα πάνω στη νέα αυτή γλώσσα.

Κεφάλαιο 1

Στοιχεία θεωρίας συνόλων

Η θεωρία συνόλων είναι το πιο διαδεδομένο σύστημα θεμελίωσης των μαθηματικών. Η σύγχρονη μελέτη της ξεκινά από τον Georg Cantor τη δεκαετία του 1870. Μετά την ανακάλυψη παραδόξων στην άτυπη θεωρία συνόλων (όπως το παράδοξο του Russell), προτάθηκαν διάφορα συστήματα αξιωμάτων, με πιο γνωστό αυτό των Zermelo-Fraenkel με την προσθήκη του αξιώματος επιλογής.

Στο κεφάλαιο αυτό θα παρουσιάσουμε κάποια στοιχεία από τη θεωρία συνόλων που θα χρειαστούν στα παρακάτω κεφάλαια. Θα αρχίσουμε με κάποιες βασικές έννοιες όπως η μερική διάταξη και τα πλήρη πλέγματα, ώστε να φτάσουμε στη διατύπωση των θεωρημάτων σταθερού σημείου των Knaster-Tarski και Kleene. Ένα πολύ καλό βιβλίο για τη θεωρία συνόλων είναι το [22].

1.1 Σύνολα

Σύνολο ονομάζουμε μια συλλογή διακριτών αντικειμένων. Τα αντικείμενα που περιλαμβάνονται στο σύνολο ονομάζονται *στοιχεία* του συνόλου. Ο συμβολισμός $x \in S$ χρησιμοποιείται για να εκφράσει ότι το στοιχείο x ανήκει στο σύνολο S , ενώ ο συμβολισμός $x \notin S$ χρησιμοποιείται για να εκφράσει ότι το στοιχείο x δεν ανήκει στο σύνολο S .

Υπάρχουν διάφοροι τρόποι για να περιγράψουμε ένα σύνολο. Για μικρά σύνολα μπορούμε να γράψουμε όλα τα στοιχεία τους, π.χ. το σύνολο $\{0, 1, 2, 3\}$ είναι το σύνολο των φυσικών μικρότερων του 4. Για μεγαλύτερα σύνολα, για παράδειγμα το σύνολο των φυσικών μικρότερων του 1000 μπορούμε είτε να γράψουμε ένα αρκετό πλήθος των στοιχείων του, όπως $\{0, 1, 2, 3, \dots, 999\}$ είτε να δώσουμε μια ιδιότητα ή συνθήκη που χαρακτηρίζει τα στοιχεία του συνόλου, όπως $\{x : 0 \leq x < 1000\}$. Το

σύνολο $\{x_1, x_2, \dots, x_n\}$ θεωρείται πεπερασμένο ενώ το $\{x_1, x_2, \dots\}$ άπειρο. Το σύνολο των φυσικών συμβολίζεται με $\mathbb{N}$, και το κενό σύνολο με το σύμβολο $\emptyset$.

Η ένωση $S_1 \cup S_2$ δύο συνόλων είναι το σύνολο των στοιχείων που ανήκουν είτε στο S_1 είτε στο S_2 , η τομή $S_1 \cap S_2$ δύο συνόλων είναι το σύνολο των στοιχείων που ανήκουν και στο S_1 και στο S_2 και η διαφορά $S_1 \setminus S_2$ δύο συνόλων είναι το σύνολο των στοιχείων που ανήκουν στο S_1 αλλά όχι στο S_2 . Ένα σύνολο S_1 είναι υποσύνολο του S_2 (γράφεται $S_1 \subseteq S_2$) όταν κάθε στοιχείο του S_1 ανήκει επίσης και στο S_2 . Δυναμοσύνολο ενός συνόλου S είναι το σύνολο όλων των υποσυνόλων του S , δηλαδή είναι $2^S = \{X : X \subseteq S\}$.

Το καρτεσιανό γινόμενο $S_1 \times S_2$ δύο συνόλων είναι το σύνολο όλων των ζευγαριών $\langle x_1, x_2 \rangle$ τέτοια ώστε $x_1 \in S_1$ και $x_2 \in S_2$. Αυτό μπορεί να γενικευθεί σε πολλά σύνολα. Οπότε, $S_1 \times S_2 \times \dots \times S_n$ το σύνολο όλων των n -άδων $\langle x_1, x_2, \dots, x_n \rangle$ τέτοια ώστε $x_i \in S_i$, $1 \leq i \leq n$. Αν $S_i = S$, $1 \leq i \leq n$, τότε συνήθως γράφουμε S^n αντί για $S \times \dots \times S$.

1.2 Διάταξη συνόλων

Έστω σύνολο S . Μια διμελής σχέση R στο S είναι ένα υποσύνολο του $S \times S$. Συνήθως γράφουμε $R(x, y)$ ή ακόμη και xRy αντί για $\langle x, y \rangle \in R$. Αν για μία σχέση $f \subseteq S_1 \times S_2$ ισχύει ότι $f(x, y), f(x, z) \implies y = z$, τότε η σχέση αυτή ονομάζεται *συνάρτηση*. Σε αυτήν την περίπτωση αντί για $f \subseteq S_1 \times S_2$ γράφουμε $f : S_1 \rightarrow S_2$ και αντί για $f(x, y)$ γράφουμε $f(x) = y$. Τελεστή ονομάζουμε μια συνάρτηση η οποία εφαρμόζεται πάνω σε μια άλλη συνάρτηση.

Ορισμός 1.1 (Σχέση μερικής διάταξης). Μια διμελής σχέση R στο S ονομάζεται *μερικής διάταξης* αν είναι:

1. ανακλαστική, δηλαδή ισχύει xRx για κάθε $x \in S$.
2. αντισυμμετρική, δηλαδή ισχύει $xRy \wedge yRx \implies x = y$, για κάθε $x, y \in S$
3. μεταβατική, δηλαδή ισχύει $xRy \wedge yRz \implies xRz$, για κάθε $x, y, z \in S$

Για παράδειγμα, αν S σύνολο και 2^S το σύνολο όλων των υποσυνόλων του S , τότε η σχέση υποσυνόλου $\subseteq$ είναι σχέση μερικής διάταξης. Στη συνέχεια θα χρησιμοποιήσουμε το σύμβολο $\leq$ για μια σχέση μερικής διάταξης.

Ορισμός 1.2 (Σχέση ολικής διάταξης). Μια διμελής σχέση R στο S ονομάζεται *ολικής διάταξης* αν είναι:

1. αντισυμμετρική, δηλαδή ισχύει $xRy \wedge yRx \implies x = y$, για κάθε $x, y \in S$
2. ολική, δηλαδή ισχύει $xRy \vee yRx$ για κάθε $x, y \in S$.
3. μεταβατική, δηλαδή ισχύει $xRy \wedge yRz \implies xRz$, για κάθε $x, y, z \in S$

Για παράδειγμα, η σχέση $\leq$ στο $\mathbb{N}$ είναι σχέση ολικής διάταξης. Κάθε σχέση ολικής διάταξης είναι και μερικής διάταξης.

1.3 Φράγματα και πλήρη πλέγματα

Ορισμός 1.3 (Άνω και κάτω φράγμα). Έστω S σύνολο με σχέση μερικής διάταξης $\leq$ και ένα υποσύνολο $X \subseteq S$.

- Το $a \in S$ είναι άνω φράγμα του X αν για κάθε $x \in X$ είναι $x \leq a$.
- Το $b \in S$ είναι κάτω φράγμα του X αν για κάθε $x \in X$ είναι $b \leq x$.

Ορισμός 1.4 (Ελάχιστο άνω και μέγιστο κάτω φράγμα). Έστω S σύνολο με σχέση μερικής διάταξης $\leq$ και ένα υποσύνολο $X \subseteq S$.

- Το $a \in S$ είναι το ελάχιστο άνω φράγμα του X αν είναι άνω φράγμα, και για κάθε άνω φράγμα a' του X είναι $a \leq a'$.
- Το $b \in S$ είναι το μέγιστο κάτω φράγμα του X αν είναι κάτω φράγμα, και για κάθε κάτω φράγμα b' του X είναι $b' \leq b$.

Το ελάχιστο άνω φράγμα του X αν υπάρχει είναι μοναδικό, συνεπώς μπορούμε να χρησιμοποιήσουμε τον συμβολισμό $\text{lub}(X)$. Επίσης, το μέγιστο κάτω φράγμα του X αν υπάρχει είναι μοναδικό, συνεπώς μπορούμε να χρησιμοποιήσουμε τον συμβολισμό $\text{glb}(X)$.

Ορισμός 1.5 (Πλήρες πλέγμα). Έστω S σύνολο με σχέση μερικής διάταξης $\leq$. Το S ονομάζεται πλήρες πλέγμα, αν για κάθε $X \subseteq S$ υπάρχουν τα $\text{lub}(X)$ και $\text{glb}(X)$.

Για παράδειγμα, αν S σύνολο και 2^S το σύνολο όλων των υποσυνόλων του S , τότε το $\langle 2^S, \subseteq \rangle$ είναι πλήρες πλέγμα, με ελάχιστο άνω φράγμα το S και μέγιστο κάτω φράγμα το $\emptyset$. Για την ακρίβεια, για κάθε $X \subseteq 2^S$, είναι $\text{lub}(X) = \bigcup_{x \in X} x$ και $\text{glb}(X) = \bigcap_{x \in X} x$. Συμβολίζουμε με $\top$ το $\text{lub}(L)$ και $\perp$ το $\text{glb}(L)$ ενός πλήρους πλέγματος L .

Πρόταση 1.1. Έστω L πλήρες πλέγμα και $A, B \subseteq L$.

1. Αν για κάθε $a \in A$ υπάρχει $b \in B$ τέτοιο ώστε $a \leq b$, τότε $\text{lub}(A) \leq \text{lub}(B)$.
2. Αν για κάθε $a \in A$ υπάρχει $b \in B$ τέτοιο ώστε $a \geq b$, τότε $\text{glb}(A) \geq \text{glb}(B)$.

Απόδειξη. Δείχνουμε για κάθε περίπτωση ξεχωριστά.

1. Για κάθε $b \in B$ είναι $b \leq \text{lub}(B)$. Επίσης, για κάθε $a \in A$ υπάρχει $b \in B$ τέτοιο ώστε $a \leq b \leq \text{lub}(B) \implies$ το $\text{lub}(B)$ είναι άνω φράγμα του $A \implies \text{lub}(A) \leq \text{lub}(B)$.
2. Για κάθε $b \in B$ είναι $b \geq \text{glb}(B)$. Επίσης, για κάθε $b \in B$ υπάρχει $a \in A$ τέτοιο ώστε $a \geq b \geq \text{glb}(B) \implies$ το $\text{glb}(B)$ είναι κάτω φράγμα του $A \implies \text{glb}(A) \geq \text{glb}(B)$.

□

Πρόταση 1.2. Έστω L πλήρες πλέγμα και $S \subseteq L$.

1. Αν $S = \bigcup_{i \in I} S_i$ τότε $\text{lub}(S) = \text{lub}\{\text{lub}(S_i) : i \in I\}$.
2. Αν $S = \bigcup_{i \in I} \bigcup_{j \in J} S_{ij}$ τότε

$$\text{lub}\{\text{lub}\{S_{ij} : j \in J\} : i \in I\} = \text{lub}\{\text{lub}\{S_{ij} : i \in I\} : j \in J\} = \text{lub}(S).$$

Απόδειξη. 1. Βλέπε [2].

2. Έστω $S_i = \bigcup_{j \in J} S_{ij}$ και $S_j = \bigcup_{i \in I} S_{ij}$. Είναι $S = \bigcup_{i \in I} S_i = \bigcup_{j \in J} S_j$. Εφαρμόζουμε το πρώτο σκέλος της πρότασης.

$$\text{lub}\{\text{lub}\{S_{ij} : j \in J\} : i \in I\} = \text{lub}\{\text{lub}(S_i) : i \in I\} = \text{lub}(S).$$

$$\text{lub}\{\text{lub}\{S_{ij} : i \in I\} : j \in J\} = \text{lub}\{\text{lub}(S_j) : j \in J\} = \text{lub}(S).$$

□

1.4 Σταθερά σημεία

Τα σταθερά σημεία παίζουν έναν πολύ σημαντικό ρόλο σε πολλές περιοχές της επιστήμης των υπολογιστών και της λογικής. Στην ενότητα αυτή θα παρουσιάσουμε δύο βασικά θεωρήματα σταθερών σημείων που χρησιμοποιούμε για να δώσουμε σημασιολογία στα λογικά προγράμματα. Αρχικά όμως θα πρέπει να ορίσουμε τις έννοιες της μονοτονικότητας, της συνέχειας και του σταθερού σημείου.

Ορισμός 1.6 (Μονοτονικότητα). Έστω $\langle L, \leq \rangle$ πλήρες πλέγμα και τελεστής $T : L \rightarrow L$. Ο T ονομάζεται *μονοτονικός*, αν για κάθε $x, y \in L$ ισχύει ότι $x \leq y \implies T(x) \leq T(y)$.

Ορισμός 1.7 (Κατευθυνόμενο σύνολο). Έστω $\langle L, \leq \rangle$ πλήρες πλέγμα και $X \subseteq L$. Το X ονομάζεται *κατευθυνόμενο*, αν κάθε πεπερασμένο υποσύνολο του X έχει άνω φράγμα στο X .

Ορισμός 1.8 (Συνέχεια). Έστω $\langle L, \leq \rangle$ πλήρες πλέγμα και τελεστής $T : L \rightarrow L$. Ο T ονομάζεται *συνεχής*, αν για κάθε κατευθυνόμενο $X \subseteq L$ ισχύει ότι $T(\text{lub}(X)) = \text{lub}(T(X))$, όπου $T(X) = \{T(x) : x \in X\}$.

Ορισμός 1.9 (Σταθερό σημείο). Έστω $\langle L, \leq \rangle$ πλήρες πλέγμα και τελεστής $T : L \rightarrow L$. Το $a \in L$ ονομάζεται *σταθερό σημείο* του T αν ισχύει ότι $T(a) = a$.

Ορισμός 1.10 (Ελάχιστο και μέγιστο σταθερό σημείο). Έστω $\langle L, \leq \rangle$ πλήρες πλέγμα και τελεστής $T : L \rightarrow L$.

- Το $a \in L$ ονομάζεται *ελάχιστο σταθερό σημείο* του T αν είναι σταθερό σημείο του T , και για κάθε σταθερό σημείο a' του T ισχύει ότι $a \leq a'$.
- Το $b \in L$ ονομάζεται *μέγιστο σταθερό σημείο* του T αν είναι σταθερό σημείο του T , και για κάθε σταθερό σημείο b' του T ισχύει ότι $b' \leq b$.

Στη συνέχεια, ακολουθεί το θεώρημα σταθερού σημείου για την περίπτωση της μονοτονικότητας.

Θεώρημα 1.1 (Knaster-Tarski). Έστω $\langle L, \leq \rangle$ πλήρες πλέγμα και μονοτονικός τελεστής $T : L \rightarrow L$. Ο T έχει ένα ελάχιστο σταθερό σημείο $\text{lfp}(T)$ και ένα μέγιστο σταθερό σημείο $\text{gfp}(T)$. Επίσης, είναι $\text{lfp}(T) = \text{glb}\{x : T(x) = x\} = \text{glb}\{x : T(x) \leq x\}$ και $\text{gfp}(T) = \text{lub}\{x : x = T(x)\} = \text{lub}\{x : x \leq T(x)\}$.

Απόδειξη. Βλεπε [32]. □

Παρατηρούμε ότι η μονοτονικότητα μας εγγυάται σταθερό σημείο. Επειδή αποδεικνύεται ότι κάθε συνεχής τελεστής είναι και μονοτονικός, θα περιμένουμε ότι η συνέχεια εκτός από σταθερό σημείο θα μας εγγυάται και κάτι παραπάνω. Αυτό είναι η δυνατότητα να προκύψει το σταθερό σημείο «κατασκευαστικά», δηλαδή εφαρμόζοντας τον T «στον εαυτό του πολλές φορές», δηλαδή:

$$\perp, T(\perp), T(T(\perp)), T(T(T(\perp))), \dots$$

Επειδή όμως σε πολλές περιπτώσεις οι φυσικοί αριθμοί «δεν φτάνουν», θα χρειαστεί να μετρήσουμε «καλύτερα» χρησιμοποιώντας την έννοια των διατακτικών αριθμών.

Διαισθητικά, οι διατακτικοί αριθμοί είναι αυτό που χρησιμοποιούμε όταν μετράμε. Κάθε διατακτικός αριθμός ορίζεται ως το σύνολο όλων των προηγούμενων του. Ο πρώτος διατακτικός είναι ο $0 = \emptyset$, ενώ οι επόμενοι είναι οι $1 = \{0\} = \{\emptyset\}$, $2 = \{0, 1\} = \{\emptyset, \{\emptyset\}\}$, $3 = \{0, 1, 2\} = \{\emptyset, \{\emptyset\}, \{\emptyset, \{\emptyset\}\}\}$ κ.ο.κ. Αυτοί είναι οι πεπερασμένοι διατακτικοί. Ο πρώτος άπειρος διατακτικός είναι ο $\omega = \{1, 2, 3, \dots\}$. Επόμενοι διατακτικοί ονομάζονται αυτοί που παράγονται από τον προηγούμενό τους $+1$ δηλαδή ισχύει $\alpha + 1 = \alpha \cup \{\alpha\}$. Αυτό ισχύει για παράδειγμα στους φυσικούς. Οι διατακτικοί για τους οποίους δεν ισχύει αυτό (εκτός από το 0) ονομάζονται οριακοί διατακτικοί, όπως για παράδειγμα ο ω . Μετά τον ω ακολουθούν οι $\omega + 1 = \omega \cup \{\omega\}$, $\omega + 2$, $\omega + 3$ μέχρι να φτάσουμε στον επόμενο οριακό διατακτικό, τον ω^2 . Συνεπώς η σειρά είναι διαισθητικά η εξής:

$$0, 1, 2, \dots, \omega, \omega + 1, \omega + 2, \dots, \omega^2, \omega^2 + 1, \dots, \omega^3, \dots, \omega^n, \dots, \omega^2, \dots, \omega^\omega, \dots$$

Στη συνέχεια θα ορίσουμε το πως γίνεται η εφαρμογή ενός μονοτονικού τελεστή στον εαυτό του α φορές, όπου α διατακτικός αριθμός.

Ορισμός 1.11. Έστω $\langle L, \leq \rangle$ πλήρες πλέγμα και μονοτονικός τελεστής $T : L \rightarrow L$. Ορίζουμε

- $T \uparrow 0 = \perp$
- $T \uparrow \alpha = \begin{cases} T(T \uparrow (\alpha - 1)), & \text{αν } \alpha \text{ επόμενος διατακτικός} \\ \text{lub}\{T \uparrow \beta : \beta < \alpha\}, & \text{αν } \alpha \text{ οριακός διατακτικός} \end{cases}$
- $T \downarrow 0 = \top$
- $T \downarrow \alpha = \begin{cases} T(T \downarrow (\alpha - 1)), & \text{αν } \alpha \text{ επόμενος διατακτικός} \\ \text{lub}\{T \downarrow \beta : \beta < \alpha\}, & \text{αν } \alpha \text{ οριακός διατακτικός} \end{cases}$

Στη συνέχεια, ακολουθεί το θεώρημα σταθερού σημείου για την περίπτωση της συνέχειας.

Θεώρημα 1.2 (Kleene). Έστω $\langle L, \leq \rangle$ πλήρες πλέγμα και συνεχής τελεστής $T : L \rightarrow L$. Τότε θα ισχύει $\text{lfp}(T) = T \uparrow \omega$.

Απόδειξη. Βλεπε [32].

□

Συνεπώς στην περίπτωση που ο T είναι συνεχής, για την εύρεση του $lfp(T)$ θα χρειαστούν το πολύ ω εφαρμογές του T , κάτι που δεν ισχύει όμως απαραίτητα για το $gfp(T)$. Υπάρχει αντιπαράδειγμα [32] για το οποίο $gfp(T) \neq T \downarrow \omega$.

Κεφάλαιο 2

Λογικός προγραμματισμός

Στο κεφάλαιο αυτό θα παρουσιάσουμε το συντακτικό και τη δηλωτική σημασιολογία των van Emden και Kowalski για τον κλασικό λογικό προγραμματισμό [46]. Η παρουσίαση δεν θα είναι εξαντλητική. Για περισσότερες λεπτομέρειες ανατρέξτε στα [32], [39], [5], [53].

2.1 Προγραμματίζοντας με λογική

Ο λογικός προγραμματισμός βασίζεται στη μαθηματική λογική και χρησιμοποιείται τόσο ως μέσο αναπαράστασης γνώσης όσο και για υπολογισμούς. Ο τρόπος προγραμματισμού γίνεται μέσω μιας σειράς δηλωτικών εκφράσεων, οι οποίες περιγράφουν τις ιδιότητες του προβλήματος και μέσω αυτών προκύπτει το αποτέλεσμα. Αυτές οι εκφράσεις, το σύνολο δηλαδή από τους κανόνες που μοντελοποιούν το πρόβλημα ονομάζονται λογικά προγράμματα.

Σύμφωνα με την πολύ γνωστή ρήση του Robert Kowalski «Αλγόριθμος = Λογική + Έλεγχος» [31], κάθε αλγόριθμος αποτελείται από δύο βασικά συστατικά, τη λογική (τί θα επιλυθεί) και τον έλεγχο (πώς θα επιλυθεί). Κάθε είδος προγραμματισμού δίνει διαφορετικό βάρος σε καθένα από τα δύο συστατικά. Στον λογικό προγραμματισμό ο προγραμματιστής ασχολείται μόνο με τη λογική, δηλαδή την περιγραφή του συστήματος ή την κωδικοποίηση της γνώσης με τη μορφή ενός συνόλου λογικών προτάσεων. Το πως θα γίνει η επίλυση αφήνεται σε έναν μηχανισμό εξαγωγής συμπερασμάτων. Ο δηλωτικός αυτός τρόπος συγγραφής των προγραμμάτων, προϋποθέτει να γνωρίζουμε τι μπορούν ή δεν μπορούν να εκφράσουν ή να υπολογίσουν τα λογικά προγράμματα, δηλαδή τι σημαίνουν. Αυτό είναι το αντικείμενο της σημασιολογίας του λογικού προγραμματισμού.

2.2 Συντακτικό

Στην ενότητα αυτή θα ορίσουμε το συντακτικό, δηλαδή τη μορφή των λογικών προγραμμάτων. Ένα λογικό πρόγραμμα διαισθητικά είναι ένα σύνολο γεγονότων και κανόνων το οποίο μπορεί να περιγράφει τη μορφή του κωδικοποιούμενου κόσμου, δηλαδή είναι μια δήλωση του τι ισχύει στον κόσμο που θέλουμε να μοντελοποιήσουμε.

Για κάθε λογικό πρόγραμμα ορίζουμε τα εξής σύνολα:

- P : σύνολο κατηγορημάτων
- F : σύνολο συναρτησιακών συμβόλων
- V : σύνολο μεταβλητών

Σε κάθε κατηγορημα ή συναρτησιακό σύμβολο k αντιστοιχεί ένας ακέραιος αριθμός $n \geq 0$ που λέγεται βαθμός του k . Το k συμβολίζεται και k/n . Τα συναρτησιακά σύμβολα βαθμού 0 λέγονται και σταθερές.

Ορισμός 2.1 (Όρος). Ένας όρος ορίζεται ως εξής:

1. Αν $X \in V$, τότε το X είναι όρος.
2. Αν $c/0 \in F$, τότε το c είναι όρος.
3. Αν $f/n \in F$ και $t_1, t_2, \dots, t_n$ είναι όροι, τότε το $f(t_1, t_2, \dots, t_n)$ είναι όρος.

Ορισμός 2.2 (Ατομικός τύπος). Ένας ατομικός τύπος (ή άτομο) ορίζεται ως εξής:

1. Αν $p/0 \in P$, τότε το p είναι άτομο.
2. Αν $p/n \in P$ και $t_1, t_2, \dots, t_n$ είναι όροι, τότε το $p(t_1, t_2, \dots, t_n)$ είναι άτομο.

Διαισθητικά, ένας όρος αναπαριστά μια οντότητα ή ένα αντικείμενο του κωδικοποιούμενου κόσμου, ενώ ένας ατομικός τύπος αναπαριστά ένα γεγονός, μια ιδιότητα, ή μια σχέση ανάμεσα σε οντότητες ή αντικείμενα του κωδικοποιούμενου κόσμου.

Ορισμός 2.3 (Τύπος). Ένας καλοσχηματισμένος τύπος (ή τύπος) ορίζεται ως εξής:

1. Αν το A είναι άτομο, τότε το A είναι τύπος.
2. Αν τα ϕ, ϕ_1, ϕ_2 είναι τύποι, τότε τα $(\neg\phi)$, $(\phi_1 \wedge \phi_2)$, $(\phi_1 \vee \phi_2)$, $(\phi_1 \leftarrow \phi_2)$, $(\phi_1 \leftrightarrow \phi_2)$ είναι τύποι.
3. Αν το ϕ είναι τύπος και $x \in V$, τότε τα $\forall x \phi$ και $\exists x \phi$ είναι τύποι.

Στους τύπους $\forall x \phi$ και $\exists x \phi$, το ϕ ονομάζεται *εμβέλεια* των ποσοδεικτών $\forall$ και $\exists$. Μια εμφάνιση κάποιας μεταβλητής αμέσως μετά τον ποσοδείκτη και μέσα στην εμβέλειά του, ονομάζεται *δεσμευμένη*, ενώ η εμφάνισή της έξω από την εμβέλεια οποιουδήποτε ποσοδείκτη, ονομάζεται *ελεύθερη*. Ένας τύπος που στερείται ελεύθερων μεταβλητών θα ονομάζεται *κλειστός τύπος*.

Ορισμός 2.4 (Κανόνας). Ένας οριστικός κανόνας (ή *κανόνας*) είναι ένας τύπος της μορφής:

$$\forall x_1 \forall x_2 \dots \forall x_s (A \vee \neg A_1 \vee \neg A_2 \vee \dots \vee \neg A_n)$$

όπου τα $A, A_1, A_2, \dots, A_n$ είναι άτομα και $x_1, x_2, \dots, x_s$ οι μεταβλητές που εμφανίζονται στα άτομα αυτά. Το παραπάνω συμβολίζεται για ευκολία ως:

$$A \leftarrow A_1 \wedge A_2 \wedge \dots \wedge A_n$$

Το A ονομάζεται *κεφαλή* και το $A_1 \wedge A_2 \wedge \dots \wedge A_n$ ονομάζεται *σώμα* του κανόνα.

Ορισμός 2.5 (Στόχος). Ένας στόχος είναι ένας τύπος της μορφής:

$$\leftarrow A_1 \wedge A_2 \wedge \dots \wedge A_n$$

δηλαδή είναι ένας κανόνας με κενή κεφαλή.

Ορισμός 2.6 (Λογικό πρόγραμμα). Ορίζουμε ως οριστικό λογικό πρόγραμμα (ή *λογικό πρόγραμμα*) ένα σύνολο οριστικών κανόνων.

Το επίθετο «οριστικός» χρησιμοποιείται για να διαχωρίσει την απλή αυτή μορφή των λογικών προγραμμάτων από άλλες επεκτάσεις τους, όπως την ύπαρξη άρνησης στα σώματα των κανόνων ή την ύπαρξη παραπάνω του ενός ατόμων στην κεφαλή του κανόνα. Δεν θα ασχοληθούμε σε αυτό το σημείο περισσότερο με αυτές τις επεκτάσεις.

Παράδειγμα 2.1 (Γενεαλογικό δέντρο).

$$P_1 = \left\{ \begin{array}{ll} \text{father(uranus, cronus),} & \text{father(cronus, zeus),} \\ \text{father(cronus, hades),} & \text{father(zeus, apollo),} \\ \text{father(zeus, ares),} & \text{father(zeus, hermes),} \\ \text{grandfather}(X, Y) \leftarrow \text{father}(X, Z) \wedge \text{father}(Z, Y), \\ \text{ancestor}(X, Y) \leftarrow \text{father}(X, Y), \\ \text{ancestor}(X, Y) \leftarrow \text{father}(X, Z) \wedge \text{ancestor}(Z, Y), \end{array} \right\}$$

Διαισθητικά, το κατηγορήμα $grandfather(X, Y)$ ισχύει αν ο X είναι παππούς του Y , π.χ. το $grandfather(cronus, hermes)$ είναι αληθές ενώ το $grandfather(cronus, hades)$ είναι ψευδές. Επίσης, το κατηγορήμα $ancestor(X, Y)$ ισχύει αν ο X είναι πρόγονος του Y , π.χ. το $ancestor(uranus, ares)$ και το $ancestor(zeus, apollo)$ είναι αληθή ενώ το $ancestor(hermes, apollo)$ είναι ψευδές.

Παράδειγμα 2.2 (Περιττοί και άρτιοι αριθμοί).

$$P_2 = \left\{ \begin{array}{l} even(0), \\ odd(s(X)) \leftarrow even(X), \\ even(s(X)) \leftarrow odd(X), \end{array} \right\}$$

Το συναρτησιακό σύμβολο s και η σταθερά 0 χρησιμοποιούνται για την αναπαράσταση των φυσικών αριθμών, π.χ. το 0 αναπαριστά τον αριθμό 0 , το $s(0)$ αναπαριστά τον αριθμό 1 (δηλαδή τον επόμενο του 0), το $s(s(0))$ αναπαριστά τον αριθμό 2 (δηλαδή τον επόμενο του επομένου του 0) κ.ο.κ.. Το κατηγορήμα $even(X)$ είναι αληθές αν ο X είναι άρτιος αριθμός ενώ το κατηγορήμα $odd(X)$ είναι αληθές αν ο X είναι περιττός αριθμός, δηλαδή τα $even(0)$, $odd(s(0))$, $even(s(s(0)))$ είναι αληθή.

2.3 Μοντελοθεωρητική σημασιολογία

Εκτός από τη συντακτική πλευρά των λογικών προγραμμάτων που εξετάσαμε στην προηγούμενη ενότητα, μας ενδιαφέρει να περιγράψουμε τι σημαίνουν τα προγράμματα αυτά. Επειδή τα λογικά προγράμματα εκφράζουν μόνο θετική γνώση και δεν δηλώνουν πότε μια σχέση δεν ισχύει, δεν είναι δυνατόν να φτιάξουμε προγράμματα με αντικρουόμενους κανόνες. Τα λογικά προγράμματα έχουν δηλαδή όπως λέμε, πάντα μοντέλο. Στην ενότητα αυτή λοιπόν θα περιγράψουμε τη μοντελοθεωρητική σημασιολογία των λογικών προγραμμάτων, θα δείξουμε δηλαδή ότι κάθε λογικό πρόγραμμα έχει ένα ελάχιστο μοντέλο.

Πρωτού όμως περάσουμε στα λογικά προγράμματα, ας ξεκινήσουμε ορίζοντας τις έννοιες της ερμηνείας, της αλήθειας και του μοντέλου για ένα σύνολο τύπων στην γενική τους μορφή.

Ορισμός 2.7 (Ερμηνεία). Μια ερμηνεία I για ένα σύνολο τύπων αποτελείται από:

1. Ένα μη κενό σύνολο D (πεδίο της ερμηνείας).
2. Μια ανάθεση σε κάθε σταθερά c ενός στοιχείου $c_I \in D$.

3. Μια ανάθεση σε κάθε συναρτησιακό σύμβολο f/n μιας συνάρτησης $f_I : D^n \rightarrow D$.
4. Μια ανάθεση σε κάθε κατηγορημα p/n μιας σχέσης p_I πάνω στο D^n .

Διαισθητικά, μια ερμηνεία περιγράφει μια «υποψήφια κατάσταση» στον κωδικοποιούμενο κόσμο. Η «κατάσταση» είναι «υποψήφια» γιατί δεν ξέρουμε αν είναι αληθής. Στη συνέχεια θα ορίσουμε πότε μια ερμηνεία είναι αληθής.

Ορισμός 2.8 (Αποτίμηση). Μια αποτίμηση σ πάνω σε μια ερμηνεία I είναι μια συνάρτηση που αντιστοιχίζει σε κάθε μεταβλητή ένα στοιχείο του D .

Ορισμός 2.9 (Ορισμός αλήθειας του Tarski). Έστω I μια ερμηνεία με πεδίο D και σ μια αποτίμηση πάνω στην I . Ένας τύπος είναι αληθής στα I, σ αν:

- είναι της μορφής $p(t_1, \dots, t_n)$ και $(\sigma(t_1), \dots, \sigma(t_n)) \in p_I$,
- είναι της μορφής $\neg \phi$ και ο ϕ είναι αληθής στα I, σ ,
- είναι της μορφής $\phi_1 \wedge \phi_2$ και οι ϕ_1, ϕ_2 είναι αληθείς στα I, σ ,
- είναι της μορφής $\phi_1 \vee \phi_2$ και ένας από τους ϕ_1, ϕ_2 είναι αληθής στα I, σ ,
- είναι της μορφής $\phi_1 \leftarrow \phi_2$ και είτε ο τύπος ϕ_2 δεν είναι αληθής στην I είτε ο ϕ_1 είναι αληθής στα I, σ ,
- είναι της μορφής $\phi_1 \leftrightarrow \phi_2$ και είτε οι ϕ_1, ϕ_2 είναι αληθείς στα I, σ είτε κανένας από τους ϕ_1, ϕ_2 δεν είναι αληθής στα I, σ ,
- είναι της μορφής $\exists x \phi$ και υπάρχει κάποιο $d \in D$ τέτοιο ώστε ο ϕ να είναι αληθής στα $I, \sigma(x|d)$, όπου $\sigma(x|d)$ η αποτίμηση σ στην οποία η μεταβλητή x έχει αντικατασταθεί με d ,
- είναι της μορφής $\forall x \phi$ και για κάθε $d \in D$ ο ϕ είναι αληθής στα $I, \sigma(x|d)$, όπου $\sigma(x|d)$ η αποτίμηση σ στην οποία η μεταβλητή x έχει αντικατασταθεί με d .

Ένας τύπος είναι ψευδής στα I, σ αν δεν είναι αληθής στα I, σ .

Ορισμός 2.10 (Μοντέλο). Έστω I μια ερμηνεία με πεδίο D και ϕ κλειστός τύπος. Η I είναι μοντέλο του ϕ αν ο ϕ είναι αληθής στην I .

Για τα λογικά προγράμματα θα εστιάσουμε σε ένα συγκεκριμένο τύπο μοντέλων, τα Herbrand μοντέλα. Η ιδέα πίσω από τα Herbrand μοντέλα είναι το να αγνοήσουμε το πραγματικό νόημα που κρύβεται πίσω από τις σταθερές και τα συναρτησιακά σύμβολα του προγράμματος. Για παράδειγμα, στο πρόγραμμα P_1 τα *apollo*, *hermes* αντιπροσωπεύουν θεούς των αρχαίων Ελλήνων ενώ στο πρόγραμμα P_2 τα $0, s(0), s(s(0)), \dots$ αντιπροσωπεύουν τους φυσικούς $0, 1, 2, \dots$. Σε κάθε περίπτωση, επικεντρώνουμε την προσοχή μας στα ίδια τα σύμβολα και όχι στο φυσικό νόημα τους.

Ορισμός 2.11 (Σύμπαν Herbrand). Έστω P λογικό πρόγραμμα. Ορίζουμε ως *σύμπαν Herbrand* U_P το σύνολο όλων των όρων που δεν περιέχουν μεταβλητές (βασικοί όροι) και που μπορούν να παραχθούν από τις σταθερές και τα συναρτησιακά σύμβολα του P (αν δεν υπάρχουν σταθερές, προσθέτουμε μια αυθαίρετη).

Ορισμός 2.12 (Βάση Herbrand). Έστω P λογικό πρόγραμμα. Ορίζουμε ως *βάση Herbrand* B_P το σύνολο όλων των ατόμων που δεν περιέχουν μεταβλητές (βασικά άτομα) και που μπορούν να παραχθούν από τα κατηγορήματα του P με ορίσματα από το U_P .

Ορισμός 2.13 (Βασικό στιγμιότυπο). Ορίζουμε ως *βασικό στιγμιότυπο* $ground(P)$ ενός λογικού προγράμματος P το (πιθανώς άπειρο) λογικό πρόγραμμα το οποίο παράγεται αντικαθιστώντας σε κάθε κανόνα του P τις μεταβλητές με στοιχεία του U_P με όλους τους δυνατούς τρόπους.

Τα παραπάνω σύνολα είναι στην ουσία μια προσπάθεια να αφαιρέσουμε τις μεταβλητές από ένα λογικό πρόγραμμα. Αν στο πρόγραμμα υπάρχουν μόνο σταθερές, τα $U_P, B_P, ground(P)$ θα είναι πεπερασμένα, ενώ στη γενική περίπτωση και τα τρία σύνολα θα είναι άπειρα.

Παράδειγμα 2.3. Για το πρόγραμμα $P = \{even(0), even(s(s(X))) \leftarrow even(X)\}$, που μοντελοποιεί τους άρτιους αριθμούς, είναι

$$U_P = \{0, s(0), s(s(0)), s(s(s(0))), \dots\}$$

$$B_P = \{even(0), even(s(0)), even(s(s(0))), even(s(s(s(0)))), \dots\}$$

$$ground(P) = \left\{ \begin{array}{c} even(0), \\ even(s(s(0))) \leftarrow even(0), \\ even(s(s(s(0)))) \leftarrow even(s(0)), \\ even(s(s(s(s(0))))) \leftarrow even(s(s(0))), \\ \dots \end{array} \right\}$$

Ορισμός 2.14 (Ερμηνεία Herbrand). Ορίζουμε ως *ερμηνεία Herbrand* I ενός προγράμματος P ένα οποιοδήποτε υποσύνολο της B_P ($I \subseteq B_P$).

Ορισμός 2.15 (Μοντέλο Herbrand). Έστω I Herbrand ερμηνεία ενός προγράμματος P . Η I είναι *μοντέλο* του P αν κάθε κανόνας του P είναι αληθής στην I .

Για τα λογικά προγράμματα ισχύει το παρακάτω θεώρημα.

Θεώρημα 2.1 (Θεώρημα τομής μοντέλων). Έστω $\mathcal{M} = \{M_1, M_2, \dots\}$ ένα μη κενό σύνολο μοντέλων ενός προγράμματος P . Τότε και η ερμηνεία $\bigcap \mathcal{M}$ είναι μοντέλο του P .

Απόδειξη. Βλέπε [32]. □

Καθώς για κάθε λογικό πρόγραμμα P ολόκληρη η B_P είναι μοντέλο του P , άρα το σύνολο των μοντέλων του P είναι μη κενό. Συνεπώς η τομή όλων των μοντέλων του P είναι μοντέλο του P . Το μοντέλο αυτό το ονομάζουμε *ελάχιστο Herbrand μοντέλο* του P και το συμβολίζουμε ως M_P .

Παράδειγμα 2.4. [53]. Έστω το λογικό πρόγραμμα

$$P = \{p(X) \leftarrow q(X), q(X) \leftarrow r(X), p(a), q(b), r(c)\}$$

Είναι $U_P = \{a, b, c\}$ και $B_P = \{p(a), p(b), p(c), q(a), q(b), q(c), r(a), r(b), r(c)\}$. Απαριθμώντας όλα τα μοντέλα του P , δηλαδή τα υποσύνολα της B_P για τα οποία όλοι οι κανόνες του P είναι αληθείς και παίρνοντας την τομή τους, καταλήγουμε στο ότι το ελάχιστο μοντέλο του P είναι το $M_P = \{p(a), p(b), p(c), q(b), q(c), r(c)\}$.

Συνεπώς, κάθε λογικό πρόγραμμα έχει ένα ελάχιστο μοντέλο, την τομή όλων των μοντέλων του. Ο τρόπος όμως που περιγράψαμε στο προηγούμενο παράδειγμα, δηλαδή η απαρίθμηση όλων των $2^{|B_P|}$ δυνατών μοντέλων δεν μπορεί να είναι η διαδικασία με την οποία βρίσκουμε τα ελάχιστα μοντέλα. Στην επόμενη ενότητα θα δείξουμε πως βρίσκουμε κατασκευαστικά το ελάχιστο μοντέλο ενός λογικού προγράμματος.

2.4 Σημασιολογία σταθερού σημείου

Στην ενότητα αυτή θα περιγράψουμε μια άλλη μεθοδολογία για τη σημασιολογία των λογικών προγραμμάτων, τη σημασιολογία σταθερού σημείου. Έτσι θα μπορούμε να έχουμε μια μέθοδο κατασκευής του ελάχιστου μοντέλου ενός λογικού προγράμματος. Θα χρησιμοποιήσουμε κάποιους ορισμούς από τη θεωρία συνόλων καθώς και το θεώρημα του σταθερού σημείου.

Πρόταση 2.1. Έστω λογικό πρόγραμμα P . Το σύνολο $\langle 2^{B_P}, \subseteq \rangle$ είναι πλήρες πλέγμα.

Ορισμός 2.16. Έστω I απειρότιμη ερμηνεία ενός προγράμματος P . Ορίζουμε τον τελεστή $T_P : 2^{B_P} \rightarrow 2^{B_P}$ ως εξής:

$$T_P(I) = \{A : (A \leftarrow A_1 \wedge A_2 \wedge \dots \wedge A_n) \in \text{ground}(P) \text{ και } \{A_1, A_2, \dots, A_n\} \subseteq I\}$$

Θεώρημα 2.2. Ο τελεστής T_P είναι μονοτονικός και συνεχής.

Απόδειξη. Βλέπε [32]. □

Ο τελεστής T_P , που ονομάζεται και τελεστής άμεσου επακόλουθου, δέχεται μια ερμηνεία και μας δίνει μια νέα ερμηνεία. Επειδή είναι μονοτονικός και συνεχής, μπορούμε να εφαρμόσουμε το θεώρημα σταθερού σημείου. Το αποτέλεσμα είναι το θεώρημα το οποίο ακολουθεί.

Θεώρημα 2.3. Έστω λογικό πρόγραμμα P και M_P το ελάχιστο μοντέλο του.

1. Το M_P είναι το ελάχιστο σταθερό σημείο του T_P , δηλαδή $M_P = \text{lfp}(T_P)$.
2. $M_P = T_P \uparrow \omega$.

Απόδειξη. Βλέπε [32]. □

Συνεπώς έχουμε στη διάθεσή μας ένα εργαλείο κατασκευής του ελάχιστου μοντέλου ενός λογικού προγράμματος.

Παράδειγμα 2.5. [53]. Έστω το λογικό πρόγραμμα

$$P = \{p(X) \leftarrow q(X), q(X) \leftarrow r(X), p(a), q(b), r(c)\}$$

Οι διαδοχικές εφαρμογές του T_P είναι οι εξής:

- $T_P \uparrow 0 = \emptyset$
- $T_P \uparrow 1 = \{p(a), q(b), r(c)\}$
- $T_P \uparrow 2 = T_P(\{p(a), q(b), r(c)\}) = \{p(a), p(b), q(b), q(c), r(c)\}$
- $T_P \uparrow 3 = T_P(\{p(a), p(b), q(b), q(c), r(c)\}) = \{p(a), p(b), p(c), q(b), q(c), r(c)\}$
- $T_P \uparrow 4 = T_P(\{p(a), p(b), p(c), q(b), q(c), r(c)\}) = \{p(a), p(b), p(c), q(b), q(c), r(c)\}$

Συνεπώς είναι $M_P = \{p(a), p(b), p(c), q(b), q(c), r(c)\}$.

Παράδειγμα 2.6. Έστω το λογικό πρόγραμμα

$$P = \{even(0), even(s(s(X))) \leftarrow even(X)\}$$

Οι διαδοχικές εφαρμογές του T_P είναι οι εξής:

- $T_P \uparrow 0 = \emptyset$
- $T_P \uparrow 1 = \{even(0)\}$
- $T_P \uparrow 2 = \{even(0), even(s(s(0)))\}$
- $T_P \uparrow 3 = \{even(0), even(s(s(0))), even(s(s(s(0))))\}$
- ...
- $T_P \uparrow \omega = \{even(s^{2i}(0)) : i \in \mathbb{N}\}$

Συνεπώς είναι $M_P = \{even(s^{2i}(0)) : i \in \mathbb{N}\}$.

2.5 Επεκτάσεις του λογικού προγραμματισμού

Καθώς ένα λογικό πρόγραμμα δεν είναι τίποτε άλλο παρά ένα σύνολο από κανόνες, μπορούν να οριστούν διάφορες επεκτάσεις του λογικού προγραμματισμού, είτε αλλάζοντας τη μορφή των κανόνων είτε εισάγοντας επιπλέον λογικούς τελεστές. Στην ενότητα αυτή θα κάνουμε μια σύντομη αναφορά σε μερικές επεκτάσεις του κλασικού λογικού προγραμματισμού. Μια λίστα από επεκτάσεις είναι η παρακάτω:

- **Διαζευκτικός λογικός προγραμματισμός** [35]. Η μορφή του κανόνα στα κλασικά (ή οριστικά) λογικά προγράμματα είναι $a \leftarrow b_1 \wedge \dots \wedge b_n$ ή ισοδύναμα $a \vee \neg b_1 \vee \dots \vee \neg b_n$. Αν επιτρέψουμε περισσότερα από ένα θετικά άτομα στην επεκτεταμένη μορφή του κανόνα θα έχουμε κανόνες της μορφής $a_1 \vee \dots \vee a_m \vee \neg b_1 \vee \dots \vee \neg b_n$ ή $a_1 \vee \dots \vee a_m \leftarrow b_1 \wedge \dots \wedge b_n$. Με αυτόν τον τρόπο εισάγουμε το στοιχείο της αβεβαιότητας αφού η ισχύς των $b_1, \dots, b_n$ ορίζει ότι θα ισχύουν κάποια από τα $a_1, \dots, a_m$.
- **Λογικός προγραμματισμός με άρνηση** [47], [16], [42]. Πολλές φορές θέλουμε να εκφράσουμε ότι κάτι ισχύει όταν κάτι άλλο δεν ισχύει. Θα έχουμε δηλαδή κανόνες της μορφής $a \leftarrow b_1 \wedge \dots \wedge b_n \wedge \sim c_1 \wedge \dots \wedge \sim c_m$. Στον λογικό προγραμματισμό με άρνηση χρησιμοποιούμε την άρνηση μέσω αποτυχίας (συμβολίζεται με $\sim$), υποθέτουμε δηλαδή ότι αν δεν μπορούμε να αποδείξουμε κάτι μέσα από το πρόγραμμά μας, τότε αυτό δεν ισχύει (υπόθεση του κλειστού κόσμου).
- **Διαζευκτικός λογικός προγραμματισμός με άρνηση** [8]. Η μορφή του κανόνα είναι $a_1 \vee \dots \vee a_m \leftarrow b_1 \wedge \dots \wedge b_n \wedge \sim c_1 \wedge \dots \wedge \sim c_m$, δηλαδή συνδυάζουμε τις

δύο παραπάνω επεκτάσεις.

- *Λογικός προγραμματισμός με περιορισμούς* [24]. Είναι μια επέκταση του κλασικού λογικού προγραμματισμού. Στα σώματα των κανόνων μπορούν να περιέχονται κάποια κατηγορήματα περιορισμών, τα οποία αντιμετωπίζονται με ειδικό τρόπο από το σύστημα εξαγωγής συμπερασμάτων, διαφορετικό από τα κανονικά κατηγορήματα, κάνοντας την εκτέλεση των προγραμμάτων πιο αποδοτική.
- *Απαγωγικός λογικός προγραμματισμός* [27]. Στην επέκταση αυτή επιτρέπουμε κανόνες της μορφής $F \leftarrow b_1 \wedge \dots \wedge b_n \wedge \sim c_1 \wedge \dots \wedge \sim c_m$. Αυτό σημαίνει ότι δεν μπορούν ταυτόχρονα τα $b_1, \dots, b_n$ να ισχύουν και τα $c_1, \dots, c_m$ να μην ισχύουν. Με αυτόν τον τρόπο μπορούμε να εκφράσουμε ότι κάτι δεν ισχύει, ή να ορίσουμε μερικώς κάποια κατηγορήματα, περιγράφοντας πότε δεν ισχύουν.
- *Επαγωγικός λογικός προγραμματισμός* [36]. Ένα επαγωγικό λογικό πρόγραμμα αποτελείται από ένα σύνολο λογικών κανόνων, ένα σύνολο ατόμων που είναι αληθή (θετικά παραδείγματα) και ένα σύνολο ατόμων που είναι ψευδή (αρνητικά παραδείγματα). Ζητούμε την εξαγωγή υποθέσεων από τις δοσμένες παρατηρήσεις, δηλαδή ένα νέο σύνολο κανόνων που μπορούμε να συμπεράνουμε από τα θετικά και αρνητικά παραδείγματα.
- *Λογικός προγραμματισμός υψηλής τάξης* [34], [9]. Σε αυτήν την επέκταση τα κατηγορήματα μπορούν να παίρνουν ως ορίσματα άλλα κατηγορήματα, και όχι μόνο όρους, όπως συμβαίνει στον κλασικό λογικό προγραμματισμό. Η ιδέα αυτή προέρχεται από τον συναρτησιακό προγραμματισμό.
- *Πιθανοτικός λογικός προγραμματισμός* [38]. Οι τιμές αληθείας σε αυτά τα προγράμματα δεν είναι οι *False*, *True*, αλλά πραγματικές τιμές στο διάστημα $[0, 1]$, και εκφράζουν την πιθανότητα να ισχύει κάτι.
- *Χρονικός λογικός προγραμματισμός* [1]. Στη χρονική λογική εισάγεται η έννοια του χρόνου, δηλαδή μια πρόταση (π.χ. η πρόταση «Πεινάω») μπορεί να είναι αληθής κάποια χρονική στιγμή αλλά ψευδής κάποια άλλη. Σε μια χρονική λογική μπορούν επομένως να εκφραστούν προτάσεις όπως: «Πεινάω πάντα», «Τελικά θα πεινάσω», ή «Θα πεινάω μέχρι να φάω κάτι».

Αυτές ήταν μερικές μόνο από τις διάφορες επεκτάσεις που έχουν προταθεί για τον λογικό προγραμματισμό.

Κεφάλαιο 3

Απειρότιμη λογική και προτιμήσεις

Στο κεφάλαιο αυτό θα περιγράψουμε μια απειρότιμη λογική, η οποία αποτελεί επέκταση της κλασικής δίτιμης λογικής και χρησιμοποιεί άπειρες τιμές αληθείας οι οποίες βρίσκονται ενδιάμεσα στις δύο ακραίες F , T . Στη συνέχεια θα χρησιμοποιήσουμε αυτήν τη λογική, ώστε να εκφράσουμε προτιμήσεις, εκμεταλλευόμενοι τη διαβάθμιση που προκύπτει στις τιμές αληθείας. Τέλος, θα εισάγουμε κάποιους τελεστές, οι οποίοι θα μας βοηθήσουν στο να συνθέτουμε νέες λογικές προτάσεις και να κάνουμε επερωτήσεις με προτιμήσεις.

3.1 Απειρότιμη λογική

Στην κλασική (δίτιμη) λογική, τα θεμέλια της οποίας είναι τόσο παλιά όσο η εποχή του Αριστοτέλη, οι προτάσεις μπορούν να παίρνουν τιμές από το σύνολο $\{F, T\}$, δηλαδή όλα τα συμπεράσματα είναι είτε ψευδή είτε αληθή. Είναι προφανές όμως ότι η ανθρώπινη λογική δεν παρουσιάζει αυτή τη μορφή, καθώς στην καθημερινή ζωή πολλές προτάσεις είναι αληθείς ή ψευδείς σε κάποιο βαθμό, ενώ άλλες δεν είναι ούτε αληθείς ούτε ψευδείς. Ένα παράδειγμα αυτών των συστημάτων λογικής είναι η τρίτιμη λογική του Kleene [29], που οι προτάσεις παίρνουν τιμές από το σύνολο $\{F, T, 0\}$, με την τιμή 0 να έχει το νόημα «δεν γνωρίζω». Η διάταξη των τιμών ως προς την αλήθεια τους είναι $F < 0 < T$.

Μια απειρότιμη λογική [42] προκύπτει από μια επέκταση της τρίτιμης λογικής, στην οποία χρησιμοποιούνται άπειρες τιμές αλήθειας «λιγότερο αληθείς» από την τιμή T και άπειρες τιμές αλήθειας «λιγότερο ψευδείς» από την τιμή F . Το σύνολο αυτό των αληθοτιμών μπορεί να εκφραστεί ως ένα άπειρο σύνολο με την παρακάτω διάταξη:

$$F_0 < F_1 < \dots < F_i < \dots < 0 < \dots < T_i < \dots < T_1 < T_0$$

Διασθητικά, οι τιμές F_0, T_0 αντιστοιχούν στις τιμές της κλασικής λογικής F, T αντίστοιχα. Οι υπόλοιπες τιμές αντιπροσωπεύουν τις ενδιάμεσες τιμές αλήθειας. Η τιμή F_0 είναι «περισσότερο ψευδής» από την τιμή F_1 , η οποία με τη σειρά της είναι «περισσότερο ψευδής» από την τιμή F_2 κ.ο.κ.. Συμμετρικά, η τιμή T_0 είναι «περισσότερο αληθής» από την τιμή T_1 , η οποία με τη σειρά της είναι «περισσότερο αληθής» από την τιμή T_2 κ.ο.κ.. Η τιμή 0 βρίσκεται «στο μέσο» του συνόλου αυτού, αφού δεν είναι ούτε αληθής ούτε ψευδής.

Ένα σημαντικό χαρακτηριστικό της λογικής αυτής είναι οι σχέσεις μεταξύ των αληθοτιμών μεταξύ τους. Κάθε αληθοτιμή λέμε ότι είναι «απείρως πιο σημαντική» από την προηγούμενή της, δηλαδή η τιμή T_k είναι «απείρως πιο αληθής» από την τιμή T_{k+1} , ενώ η τιμή F_k είναι «απείρως πιο ψευδής» από την τιμή F_{k+1} . Αυτό σημαίνει ότι η διάταξη των φυσικών αριθμών δεν είναι μια καλή αναλογία για να καταλάβουμε τη διάταξη των αληθοτιμών. Δηλαδή, για κάθε $a, b \in \mathbb{N}^+$ με $a < b$, μπορούμε να προσθέσουμε το a στον εαυτό του κάποιες φορές, ώστε να προκύψει ότι $a + a + \dots + a > b$. Στην απειρότιμη λογική όμως κάτι τέτοιο δεν μπορεί να συμβεί. Διαισθητικά, δεν μπορούν «πολλές τιμές» του T_1 να αθροιστούν με κάποιο τρόπο ώστε να περάσουμε στην επόμενη αληθοτιμή T_0 .

Η επέκταση αυτή στις τιμές αληθείας έχει άμεση σχέση με την έννοια των απειροστών στη μη τυπική ανάλυση [28]. Απειροστό ονομάζουμε μια ποσότητα ϵ για την οποία ισχύει ότι $-a < \epsilon < a$, για κάθε θετικό πραγματικό a . Διαισθητικά, η ποσότητα ϵ είναι μικρότερη από κάθε θετικό μη μηδενικό πραγματικό αριθμό, αλλά ταυτόχρονα μεγαλύτερη από το μηδέν. Άμεση συνέπεια του παραπάνω είναι ότι για $a, b, r, s \in \mathbb{R}^+$ με $r < s$ τότε θα ισχύει $a \cdot \epsilon^r < b \cdot \epsilon^s$, ανεξάρτητα με τη σχέση των a, b . Αναλογικά, το άπειρο σύνολο τιμών αληθείας της απειρότιμης λογικής προκύπτει από τον πολλαπλασιασμό των τιμών T, F της δίτιμης λογικής με δυνάμεις του ϵ . Είναι δηλαδή $F_k = \epsilon^k F$ και $T_k = \epsilon^k T$. Επίσης, είναι $\lim_{k \rightarrow \infty} \epsilon^k F = \lim_{k \rightarrow \infty} \epsilon^k T = 0$. Ο πολλαπλασιασμός μιας τιμής αληθείας με ϵ εκφράζει ότι η τιμή $\epsilon^k T$ είναι «απείρως πιο αληθής» από την τιμή $\epsilon^{k+1} T$, ενώ η τιμή $\epsilon^k F$ είναι «απείρως πιο ψευδής» από την τιμή $\epsilon^{k+1} F$. Ας συνοψίσουμε τα παραπάνω στους επόμενους ορισμούς.

Ορισμός 3.1 (Σύνολο αληθοτιμών). Ορίζουμε ως σύνολο αληθοτιμών το άπειρο σύνολο

$$\mathbb{V} = \{F_0, F_1, \dots, F_i, \dots, 0, \dots, T_i, \dots, T_1, T_0\}$$

για το οποίο ισχύει $\epsilon^k F = F_k$, $\epsilon^k T = T_k$ και $\lim_{k \rightarrow \infty} \epsilon^k F = \lim_{k \rightarrow \infty} \epsilon^k T = 0$. Το σύνολο είναι πλήρως διατεταγμένο με την παρακάτω διάταξη:

$$F_0 < F_1 < \dots < F_i < \dots < 0 < \dots < T_i < \dots < T_1 < T_0$$

Ορισμός 3.2 (Τάξη αληθοτιμής). Ορίζουμε ως τάξη μιας αληθοτιμής το: $\text{ord}(T_i) = i$,

$$\text{ord}(F_i) = i \text{ και } \text{ord}(0) = +\infty.$$

Στη συνέχεια θα επαναορίσουμε τους λογικούς τελεστές $\vee, \wedge$ και $\neg$ της κλασικής λογικής και στην απειρότιμη λογική, οι οποίοι όπως αναμένεται θα έχουν παρόμοια σημασία.

Ορισμός 3.3 (Λογικοί τελεστές). Έστω A, B προτάσεις της απειρότιμης λογικής. Είναι:

- $A \wedge B = \min\{A, B\}$
- $A \vee B = \max\{A, B\}$
- $\neg A = \begin{cases} T_i, & \text{αν } A = F_i \\ F_i, & \text{αν } A = T_i \\ 0, & \text{αν } A = 0 \end{cases}$

Ο ορισμός για τους τελεστές ισχύει και στη δίτιμη λογική, αν θεωρήσουμε (για τους τελεστές $\wedge, \vee$) τη διάταξη ως προς την αλήθεια $F < T$. Επίσης, οι γνωστές ιδιότητες από την άλγεβρα Boole που ισχύουν για τους τελεστές $\wedge, \vee$, μεταφέρονται και στην απειρότιμη λογική:

Ταυτοδυναμία

$$A \wedge A = A$$

$$A \vee A = A$$

Αντιμεταθετικότητα

$$A \wedge B = B \wedge A$$

$$A \vee B = B \vee A$$

Προσεταιριστικότητα

$$(A \wedge B) \wedge C = A \wedge (B \wedge C)$$

$$(A \vee B) \vee C = A \vee (B \vee C)$$

Επιμεριστικότητα

$$A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$$

$$A \vee (B \wedge C) = (A \vee B) \wedge (A \vee C)$$

3.2 Προτιμήσεις στην απειρότιμη λογική

Στην απειρότιμη λογική υπάρχει μια διαβάθμιση στις τιμές αλήθειας. Διαισθητικά, είναι σαν να «ξεθωριάζει» η επίδραση των ακραίων τιμών στις αληθοτιμές όσο πλησιάζουμε προς το κέντρο. Μπορούμε να εκμεταλλευτούμε τη διαβάθμιση αυτή ώστε να μπορέσουμε να εκφράσουμε προτιμήσεις στην απειρότιμη λογική.

Έστω δύο γεγονότα με την τιμή του πρώτου να είναι F_0 , ενώ του δεύτερου F_1 . Αν δούμε τη σχέση των αληθοτιμών τους από άποψη προτιμήσεων, το δεύτερο γεγονός είναι πιο προτιμητέο, γιατί είναι λιγότερο ψευδές από το πρώτο (και άρα μας ενοχλεί λιγότερο η απουσία του). Αντίστοιχα, έστω δύο γεγονότα με την τιμή του πρώτου να είναι T_0 , ενώ του δεύτερου T_1 . Αν δούμε τη σχέση των αληθοτιμών τους από άποψη προτιμήσεων, το δεύτερο γεγονός είναι λιγότερο προτιμητέο, γιατί είναι λιγότερο αληθές από το πρώτο (γιατί ίσως δεν πληρεί κάποια από τα κριτήρια που έχουμε θέσει). Συνεπώς, η διάταξη που έχουμε θέσει στις αληθοτιμές, μπορεί να θεωρηθεί ως διάταξη προτίμησης, δηλαδή αν $A, B \in \mathbb{V}$ και $A < B$, τότε το A είναι λιγότερο προτιμητέο από το B .

Κάποιες ενδιαφέρουσες ιδιότητες της σχέσης $<$ είναι οι εξής:

1. Αν $p, q \in \mathbb{V}$ ισχύει ότι $p > q \rightarrow \neg(q > p)$.
2. Αν $p, q \in \mathbb{V}$ ισχύει ακριβώς ένα από τα $p < q$, $q < p$, $p = q$.
3. Αν $p, q, r \in \mathbb{V}$ ισχύει ότι $(p < q) \wedge (q < r) \rightarrow (p < r)$.
4. Αν $a < b$ τότε $F_a < F_b$ και $T_b < T_a$.
5. $F_a < T_a$, για κάθε a .

Στη συνέχεια θα εξετάσουμε κατά πόσον η σχέση αυτή συμπεριφέρεται καλά σαν μια σχέση προτίμησης, με τη βοήθεια των παραπάνω ιδιοτήτων της σχέσης $<$.

Η σχέση $<$ είναι αντισυμμετρική, δηλαδή $(p > q) \rightarrow \neg(q > p)$. Η αντισυμμετρικότητα είναι μια σχεδόν προφανής ιδιότητα που έχει μια σχέση προτίμησης, αφού κάποιος που προτιμάει το p από το q δεν γίνεται ταυτόχρονα να προτιμάει το q από το p .

Η σχέση $<$ είναι ολική, δηλαδή για κάθε p, q ισχύει ακριβώς ένα από τα $p < q$, $q < p$, $p = q$, δηλαδή θεωρούμε ότι μεταξύ κάθε δύο γεγονότων p, q υπάρχει πάντα σύγκριση ως προς την προτίμηση. Σε ορισμένες περιπτώσεις όμως δύο αντικείμενα δεν μπορούν να συγκριθούν γιατί είναι τόσο ανόμοια ώστε να μην υπάρχει μέτρο σύγκρισης, όπως το αν προτιμάμε μια συμφωνία του Brahms ή μια βελγική μπύρα. Παρόλα αυτά, η ιδιότητα αυτή θεωρείται συνήθως δεδομένη στα συστήματα προτιμήσεων γιατί είναι χρήσιμη στις περισσότερες εφαρμογές.

Η σχέση $<$ είναι μεταβατική, δηλαδή $(p > q) \wedge (q > r) \rightarrow (p > r)$. Η μεταβατικότητα είναι μια λογική ιδιότητα που θα θέλαμε να έχει μια σχέση προτίμησης, αφού αν κάποιος προτιμάει το p από το q και το q από το r , προφανώς θα προτιμάει και το p από το r . Παρόλα αυτά, ενώ η ιδιότητα αυτή έχει κάποια καλά στοιχεία είναι σχετικά αμφιλεγόμενη σε δύο σημεία τα οποία πρέπει και να εξετάσουμε.

Το πρώτο αμφιλεγόμενο σημείο της μεταβατικότητας είναι το ότι μπορεί να οδη-

γήσει σε κάποια παράδοξα [23]. Ας θεωρήσουμε ένα σύνολο 1000 φλυτζανιών καφέ, το C_0 που να μην έχει καθόλου ζάχαρη, το C_1 ένα κόκκο ζάχαρης, το C_2 δύο κόκκους κ.ο.κ.. Παρόλο που για κάθε i προφανώς $C_i = C_{i+1}$ (αφού κάποιος δεν μπορεί να καταλάβει τη διαφορά), είναι $C_{999} \neq C_0$. Για να αποφύγουμε αυτό το παράδοξο, στο σύστημα προτιμήσεων μας θα θεωρήσουμε ότι οι προτιμήσεις μας είναι διακριτές (και όχι συνεχείς όπως φαίνεται στο παράδειγμα), πράγμα που σημαίνει ότι σε αυτήν την περίπτωση θα είναι $C_i \neq C_{i+1}$.

Το δεύτερο αμφιλεγόμενο σημείο της μεταβατικότητας είναι το ότι δεν μπορούμε να έχουμε κυκλικές προτιμήσεις. Για παράδειγμα κάποιος πρέπει να διαλέξει τα ρούχα A, B, C που ανά δύο συγκρίνονται ως εξής: $(A > B) \wedge (B > C) \wedge (C > A)$. Αν στο κατάστημα ήταν διαθέσιμα μόνο τα A, B , τότε προφανώς θα προτιμούσε το A . Αν όμως υπάρχουν και τα τρία, ο πελάτης δεν ξέρει ποιο να διαλέξει. Στο σύστημα προτιμήσεων μας θα μπορούσαμε σε τέτοιες περιπτώσεις να εκμεταλλευτούμε την τιμή αληθείας 0 του συνόλου αληθοτιμών.

Για τη σχέση $<$ ισχύει επίσης και ότι αν $a < b$ τότε $F_a < F_b$ και $T_b < T_a$, δηλαδή για δυο αληθή γεγονότα p, q (δηλαδή είναι $p, q > 0$) θα ισχύει ότι $p > q \implies \neg q > \neg p$. Αυτό διαισθητικά σημαίνει ότι αν το p είναι προτιμότερο του q , τότε η απουσία του q είναι προτιμότερη από την απουσία του p . Επίσης, από τις σχέσεις $p > q$ και $\neg q > \neg p$ προκύπτει ότι $\min\{p, \neg q\} > \min\{\neg p, q\} \iff (p \wedge \neg q) > (\neg p \wedge q)$. Αυτό σημαίνει διαισθητικά ότι όταν κάποιος προτιμάει το p από το q , σε περίπτωση που είναι αναγκασμένος να «χάσει» το ένα από τα δύο, θα επιλέξει να χάσει το q . Για παράδειγμα, όταν ένας ποδοσφαιριστής δηλώνει ότι «προτιμάει να κερδίζει η ομάδα του παρά να σκοράρει», εννοεί ότι «είναι καλύτερα να κερδίζει η ομάδα του και να μη σκοράρει, παρά να χάνει η ομάδα του και να σκοράρει». Συνεπώς αυτή η ιδιότητα της σχέσης $<$ είναι διαισθητικά συμβατή με μια σχέση προτίμησης.

Για τη σχέση $<$ ισχύει τέλος η ιδιότητα $F_a < T_a$, για κάθε a . Αυτή η ιδιότητα έχει άμεση σχέση με την έννοια του «καλού» και του «κακού» γεγονότος. Ένα γεγονός θα λέγεται «καλό» αν ισχύει $p > \neg p$, δηλαδή αν είναι προτιμότερη η ύπαρξη από την απουσία του. Ομοίως, ένα γεγονός θα λέγεται «κακό» αν ισχύει $\neg p > p$, δηλαδή αν είναι προτιμότερη η απουσία από την ύπαρξη του. Η παραπάνω ιδιότητα διαισθητικά λοιπόν λέει ότι ένα καλό γεγονός θα πρέπει να πραγματοποιείται ενώ ένα κακό όχι. Αν για ένα γεγονός p είναι $p = T_a$ τότε $T_a > F_a \implies T_a > \neg T_a \implies p > \neg p$, άρα το γεγονός αυτό είναι καλό. Αν για ένα γεγονός p είναι $p = F_a$ τότε $T_a > F_a \implies \neg F_a > F_a \implies \neg p > p$, άρα το γεγονός αυτό είναι κακό. Συνεπώς και αυτή η ιδιότητα της σχέσης $<$ είναι διαισθητικά συμβατή με μια σχέση προτίμησης.

Στην ενότητα αυτή επιχειρηματολογήσαμε για το ότι μπορούμε να εκφράσουμε προτιμήσεις στην απειρότιμη λογική, δεδομένου ότι οι προτιμήσεις που θέλουμε να εκφράσουμε έχουν τις εξής ιδιότητες: ότι οι προτιμήσεις που θεωρούμε είναι διακριτές

και όχι συνεχείς, και ότι κάθε γεγονός συγκρίνεται ως προς την προτίμηση του σε σχέση με τα υπόλοιπα γεγονότα. Αν πληρούνται τα παραπάνω κριτήρια, τότε η σχέση $<$ στην απειρότιμη λογική συμπεριφέρεται καλά ως σχέση προτίμησης.

3.3 Τελεστές προτίμησης

Στην ενότητα αυτή θα ορίσουμε τους τελεστές προτίμησης μ, ω , αρχικά τυπικά και στη συνέχεια εξετάζοντας το διαισθητικό τους νόημα.

Όπως φάνηκε στην προηγούμενη ενότητα, για να μπορέσουμε να εκφράσουμε και να διαχειριστούμε προτιμήσεις στην απειρότιμη λογική, χρειάζεται να μπορούμε να μεταβάλλουμε την αληθοτιμή μιας λογικής πρότασης, πολλαπλασιάζοντας τη με την ποσότητα ϵ . Ο απλός πολλαπλασιασμός με ϵ όμως δεν αρκεί, γιατί έχει τον περιορισμό ότι μεταβάλλει με συμμετρικό τρόπο την αληθοτιμή μιας πρότασης. Για το λόγο αυτό θα ορίσουμε δύο τελεστές προτίμησης μ και ω οι οποίοι θα έχουν το νόημα «προαιρετικά» και «εναλλακτικά» αντίστοιχα [3], [49].

Ορισμός 3.4 (Τελεστές προτίμησης). [3]. Έστω $A \in \mathbb{V}$. Είναι:

- $\mu A = (A \vee \epsilon A)$
- $\omega A = (A \wedge \epsilon A)$

Πρωτού εξετάσουμε το διαισθητικό νόημα των τελεστών προτίμησης, ας δείξουμε κάποιες ιδιότητες για τους τελεστές αυτούς. Αρχικά θα δείξουμε ότι οι τελεστές μ, ω είναι επιμεριστικοί ως προς τους $\wedge, \vee$.

Πρόταση 3.1. Έστω $A, B \in \mathbb{V}$. Είναι

$$\begin{aligned} \epsilon(A \wedge B) &= \epsilon A \wedge \epsilon B & \epsilon(A \vee B) &= \epsilon A \vee \epsilon B \\ \mu(A \wedge B) &= \mu A \wedge \mu B & \mu(A \vee B) &= \mu A \vee \mu B \\ \omega(A \wedge B) &= \omega A \wedge \omega B & \omega(A \vee B) &= \omega A \vee \omega B \end{aligned}$$

Απόδειξη. Θα δείξουμε τις ιδιότητες για κάθε τελεστή ξεχωριστά.

- $\epsilon(A \wedge B) = \epsilon(\min\{A, B\}) = \min\{\epsilon A, \epsilon B\} = \epsilon A \wedge \epsilon B$.
- $\epsilon(A \vee B) = \epsilon(\max\{A, B\}) = \max\{\epsilon A, \epsilon B\} = \epsilon A \vee \epsilon B$.
- Έστω $A \leq B$. Είναι $\mu(A \wedge B) = \mu A$. Επίσης, $\mu A \wedge \mu B = (A \vee \epsilon A) \wedge (B \vee \epsilon B) = (A \wedge B) \vee (A \wedge \epsilon B) \vee (\epsilon A \wedge B) \vee (\epsilon A \wedge \epsilon B) = A \vee A \vee \epsilon A \vee \epsilon A = A \vee \epsilon A = \mu A$.

- $\mu(A \vee B) = (A \vee B) \vee \epsilon(A \vee B) = (A \vee B) \vee (\epsilon A \vee \epsilon B) = (A \vee \epsilon A) \vee (B \vee \epsilon B) = \mu A \vee \mu B.$
- $\omega(A \wedge B) = (A \wedge B) \wedge \epsilon(A \wedge B) = (A \wedge B) \wedge (\epsilon A \wedge \epsilon B) = (A \wedge \epsilon A) \wedge (B \wedge \epsilon B) = \omega A \wedge \omega B.$
- Έστω $A \leq B$. Είναι $\omega(A \vee B) = \omega B$. Επίσης $\omega A \vee \omega B = (A \wedge \epsilon A) \vee (B \wedge \epsilon B) = (A \vee B) \wedge (A \vee \epsilon B) \wedge (\epsilon A \vee B) \wedge (\epsilon A \vee \epsilon B) = B \wedge \epsilon B \wedge B \wedge \epsilon B = B \wedge \epsilon B = \omega B.$

□

Στη συνέχεια θα εξετάσουμε ποιο είναι το αποτέλεσμα της εφαρμογής των μ και ω στον εαυτό τους πολλές φορές. Θα δείξουμε ότι οι τελεστές μ, ω είναι αθροιστικοί.

Πρόταση 3.2. [3]. Έστω $A \in \mathbb{V}$. Για $k \geq 1$ έχουμε:

- $\mu^k A = (A \vee \epsilon^k A)$
- $\omega^k A = (A \wedge \epsilon^k A)$

Απόδειξη. Με επαγωγή στο k .

Για $k = 1$ είναι $\mu A = (A \vee \epsilon A)$, αληθές από ορισμό. Έστω $\mu^k A = (A \vee \epsilon^k A)$. Είναι $\mu^{k+1} A = \mu \mu^k A = \mu(A \vee \epsilon^k A) = (A \vee \epsilon^k A) \vee \epsilon(A \vee \epsilon^k A) = A \vee \epsilon A \vee \epsilon^k A \vee \epsilon^{k+1} A = A \vee \epsilon^{k+1} A$, γιατί $\min\{A, \epsilon A, \epsilon^k A, \epsilon^{k+1} A\} = \min\{A, \epsilon^{k+1} A\}$.

Για $k = 1$ είναι $\omega A = (A \wedge \epsilon A)$, αληθές από ορισμό. Έστω $\omega^k A = (A \wedge \epsilon^k A)$. Είναι $\omega^{k+1} A = \omega \omega^k A = \omega(A \wedge \epsilon^k A) = (A \wedge \epsilon^k A) \wedge \epsilon(A \wedge \epsilon^k A) = A \wedge \epsilon A \wedge \epsilon^k A \wedge \epsilon^{k+1} A = A \wedge \epsilon^{k+1} A$, γιατί $\max\{A, \epsilon A, \epsilon^k A, \epsilon^{k+1} A\} = \max\{A, \epsilon^{k+1} A\}$. □

Στη συνέχεια, θα δείξουμε μια σχέση δυϊκότητας που υπάρχει ανάμεσα στους τελεστές μ και ω , και που προκύπτει από την εφαρμογή του ενός τελεστή στον άλλον.

Πρόταση 3.3. [3]. Έστω $A \in \mathbb{V}$. Είναι $\mu^k \omega^k A = \omega^k \mu^k A = \epsilon^k A$.

Απόδειξη. Αρχικά δείχνουμε $\mu^k \omega^k A = \epsilon^k A$ και στη συνέχεια $\omega^k \mu^k A = \epsilon^k A$.

Είναι $\mu^k \omega^k A = \mu^k (A \wedge \epsilon^k A) = (A \wedge \epsilon^k A) \vee \epsilon^k (A \wedge \epsilon^k A) = (A \wedge \epsilon^k A) \vee (\epsilon^k A \wedge \epsilon^{2k} A) = \max\{\min\{A, \epsilon^k A\}, \min\{\epsilon^k A, \epsilon^{2k} A\}\}$. Αν $A \leq \epsilon^k A$, τότε θα είναι και $\epsilon^k A \leq \epsilon^{2k} A$, άρα $\mu^k \omega^k A = \min\{\epsilon^k A, \epsilon^{2k} A\} = \epsilon^k A$. Αν $A > \epsilon^k A$, τότε θα είναι και $\epsilon^k A > \epsilon^{2k} A$, άρα $\mu^k \omega^k A = \min\{A, \epsilon^k A\} = \epsilon^k A$.

Είναι $\omega^k \mu^k A = \omega^k (A \vee \epsilon^k A) = (A \vee \epsilon^k A) \wedge \epsilon^k (A \vee \epsilon^k A) = (A \vee \epsilon^k A) \wedge (\epsilon^k A \vee \epsilon^{2k} A) = \min\{\max\{A, \epsilon^k A\}, \max\{\epsilon^k A, \epsilon^{2k} A\}\}$. Αν $A \leq \epsilon^k A$, τότε θα είναι και $\epsilon^k A \leq \epsilon^{2k} A$, άρα $\omega^k \mu^k A = \max\{A, \epsilon^k A\} = \epsilon^k A$. Αν $A > \epsilon^k A$, τότε θα είναι και $\epsilon^k A > \epsilon^{2k} A$, άρα $\omega^k \mu^k A = \max\{\epsilon^k A, \epsilon^{2k} A\} = \epsilon^k A$. □

Τέλος, θα εξετάσουμε την επίδραση των τελεστών προτίμησης σε λογικές προτάσεις ανάλογα με την τιμή αληθείας τους, ώστε τελικά να μπορέσουμε να επιβεβαιώσουμε το διαισθητικό νόημα του κάθε τελεστή.

Πρόταση 3.4. [3]. Για τους τελεστές μ και ω , για $x \geq 0$ και $k \geq 1$ ισχύει:

A	F_i	0	T_i
$\mu^k A$	F_{i+k}	0	T_i
$\omega^k A$	F_i	0	T_{i+k}

Απόδειξη. $\mu^k 0 = (0 \vee \epsilon^k 0) = 0$ και $\omega^k 0 = (0 \wedge \epsilon^k 0) = 0$.

$$\mu^k A = (A \vee \epsilon^k A) = \max\{A, \epsilon^k A\} \Rightarrow \begin{cases} \mu^k T_i = \max\{T_i, T_{i+k}\} = T_i \\ \mu^k F_i = \max\{F_i, F_{i+k}\} = F_{i+k} \end{cases}$$

$$\omega^k A = (A \wedge \epsilon^k A) = \min\{A, \epsilon^k A\} \Rightarrow \begin{cases} \omega^k T_i = \min\{T_i, T_{i+k}\} = T_{i+k} \\ \omega^k F_i = \min\{F_i, F_{i+k}\} = F_i \end{cases} \quad \square$$

Παρατηρούμε ότι η πρόταση μA είναι το ίδιο αληθής με την A αν η A είναι αληθής, ενώ είναι λιγότερο ψευδής από την A αν η A είναι ψευδής. Επίσης, η πρόταση ωA είναι λιγότερο αληθής από την A αν η A είναι αληθής, ενώ είναι το ίδιο ψευδής με την A αν η A είναι ψευδής. Μπορούμε σε αυτό το σημείο να επιβεβαιώσουμε το διαισθητικό νόημα «προαιρετικά» και «εναλλακτικά» για τους τελεστές μ και ω αντίστοιχα. Η πρόταση «προαιρετικά A » είναι το ίδιο αληθής με την A , αλλά λιγότερο ψευδής από την A , γιατί εφόσον είναι προαιρετική επιλογή δεν θα ενοχλεί τόσο η απουσία της σε σχέση με την απουσία της πρωτεύουσας επιλογής. Επίσης, η πρόταση «εναλλακτικά A » είναι το ίδιο ψευδής με την A , αλλά λιγότερο αληθής από την A , γιατί εφόσον είναι εναλλακτική επιλογή η παρουσία της δεν θα είναι τόσο αρεστή όσο η πρωτεύουσα επιλογή.

3.4 Λογικές προτάσεις με προτιμήσεις

Στην προηγούμενη ενότητα ορίσαμε τους τελεστές προτίμησης μ και ω . Στη συνέχεια θα δούμε πως μπορούμε να χτίσουμε λογικές προτάσεις και επερωτήσεις με προτιμήσεις με τους τελεστές αυτούς.

Ο τελεστής μ έχει τη σημασία «προαιρετικά» και χρησιμοποιείται σε συνδυασμό με τον τελεστή $\wedge$, με σκοπό τη σύζευξη προτάσεων, κάποιες από τις οποίες απαραίτητες, και κάποιες προτιμητέες. Έστω η παρακάτω πρόταση:

$$A_0 \wedge \mu A_1 \wedge \mu^2 A_2 \wedge \cdots \wedge \mu^n A_n$$

Σε αυτήν την πρόταση η A_0 είναι απαραίτητη, ενώ οι υπόλοιπες είναι απλά προτιμητέες, κάθε μια σε μικρότερο βαθμό από όλες τις προηγούμενες. Αν όλες οι $A_0, A_1, \dots, A_n$ είναι αληθείς τότε και η πρόταση είναι αληθής. Αν κάποια όμως είναι ψευδής (έστω η A_k) τότε η πρόταση είναι και αυτή ψευδής, αλλά με μια τιμή F_k . Έτσι, αν είναι ψευδής η A_0 η πρόταση είναι αρκετά πιο ψευδής από το αν είναι ψευδής η πρόταση A_n .

Ο τελεστής ω έχει τη σημασία «εναλλακτικά» και χρησιμοποιείται σε συνδυασμό με τον τελεστή $\vee$, με σκοπό τη διάζευξη προτάσεων, κάποιες από τις οποίες πρωτεύουσες, και κάποιες εναλλακτικές. Έστω η παρακάτω πρόταση:

$$A_0 \vee \omega A_1 \vee \omega^2 A_2 \vee \cdots \vee \omega^n A_n$$

Σε αυτήν την πρόταση η A_0 είναι η πρώτη επιλογή, ενώ οι υπόλοιπες είναι εναλλακτικές, κάθε μια σε μικρότερο βαθμό από όλες τις προηγούμενες. Αν όλες οι $A_0, A_1, \dots, A_n$ είναι ψευδείς τότε και η πρόταση είναι ψευδής. Αν κάποια όμως είναι αληθής (έστω η A_k) τότε η πρόταση είναι και αυτή αληθής, αλλά με μια τιμή T_k . Έτσι, αν είναι αληθής η A_0 η πρόταση είναι αρκετά πιο αληθής από το αν είναι αληθής η πρόταση A_n .

Η συμπεριφορά των τελεστών μ και ω μπορεί να φανεί διαισθητικά στα παρακάτω παραδείγματα, στα οποία μπορούμε να επιβεβαιώσουμε τα παραπάνω.

Αρχικά, ακολουθεί ο πίνακας αλήθειας για τις προτάσεις $A \wedge \mu B$ και $A \vee \omega B$ σε σύγκριση με τις προτάσεις $A \wedge B$ και $A \vee B$ (που είναι οι αντίστοιχες προτάσεις χωρίς προτιμήσεις) δεδομένου ότι τα A, B παίρνουν τις τιμές F_0 ή T_0 . Παρατηρούμε ότι ο μ επιδρά στις ψευδείς τιμές και ο ω στις αληθείς τιμές.

A	B	μB	ωB	$A \wedge B$	$A \wedge \mu B$	$A \vee B$	$A \vee \omega B$
F_0	F_0	F_1	F_0	F_0	F_0	F_0	F_0
F_0	T_0	T_0	T_1	F_0	F_0	T_0	T_1
T_0	F_0	F_1	F_0	F_0	F_1	T_0	T_0
T_0	T_0	T_0	T_1	T_0	T_0	T_0	T_0

Στη συνέχεια, ακολουθεί μια πιο γενική μορφή του παραπάνω πίνακα αλήθειας. Εδώ τα A, B παίρνουν οποιεσδήποτε τιμές F_a ή T_b .

A	B	μB	ωB	$A \wedge B$	$A \wedge \mu B$	$A \vee B$	$A \vee \omega B$
F_a	F_b	F_{b+1}	F_b	$F_{\min\{a,b\}}$	$F_{\min\{a,b+1\}}$	$F_{\max\{a,b\}}$	$F_{\max\{a,b\}}$
F_a	T_b	T_b	T_{b+1}	F_a	F_a	T_b	T_{b+1}
T_a	F_b	F_{b+1}	F_b	F_b	F_{b+1}	T_a	T_a
T_a	T_b	T_b	T_{b+1}	$T_{\max\{a,b\}}$	$T_{\max\{a,b\}}$	$T_{\min\{a,b\}}$	$T_{\min\{a,b+1\}}$

Στη συνέχεια, ακολουθεί ο πίνακας αλήθειας για τις προτάσεις $A_0 \wedge \mu A_1 \wedge \mu^2 A_2$ και $A_0 \vee \omega A_1 \vee \omega^2 A_2$, δεδομένου ότι τα A_0, A_1, A_2 έχουν τιμές F_0 ή T_0 .

A_0	A_1	A_2	μA_1	ωA_1	$\mu^2 A_2$	$\omega^2 A_2$	$A_0 \wedge \mu A_1 \wedge \mu^2 A_2$	$A_0 \vee \omega A_1 \vee \omega^2 A_2$
F_0	F_0	F_0	F_1	F_0	F_2	F_0	F_0	F_0
F_0	F_0	T_0	F_1	F_0	T_0	T_2	F_0	T_2
F_0	T_0	F_0	T_0	T_1	F_2	F_0	F_0	T_1
F_0	T_0	T_0	T_0	T_1	T_0	T_2	F_0	T_1
T_0	F_0	F_0	F_1	F_0	F_2	F_0	F_1	T_0
T_0	F_0	T_0	F_1	F_0	T_0	T_2	F_1	T_0
T_0	T_0	F_0	T_0	T_1	F_2	F_0	F_2	T_0
T_0	T_0	T_0	T_0	T_1	T_0	T_2	T_0	T_0

Ένα σημαντικό πλεονέκτημα των τελεστών προτίμησης είναι το ότι από την τιμή του αποτελέσματος μπορούμε να αντλήσουμε πληροφορίες για τους επιμέρους στόχους της επερώτησης. Στο παραπάνω παράδειγμα, αν στην επερώτηση $A_0 \wedge \mu A_1 \wedge \mu^2 A_2$ πάρουμε απάντηση F_0 τότε γνωρίζουμε ότι ο στόχος A_0 απέτυχε, αν πάρουμε απάντηση F_1 τότε ο στόχος A_1 απέτυχε (ενώ ο A_0 πέτυχε), και αν πάρουμε απάντηση F_2 τότε ο στόχος A_2 απέτυχε (ενώ οι προηγούμενοι πέτυχαν). Συμμετρικά, αν στην επερώτηση $A_0 \vee \omega A_1 \vee \omega^2 A_2$ πάρουμε απάντηση T_0 τότε γνωρίζουμε ότι ο στόχος A_0 πέτυχε, αν πάρουμε απάντηση T_1 τότε ο στόχος A_1 πέτυχε (ενώ ο A_0 απέτυχε), και αν πάρουμε απάντηση T_2 τότε ο στόχος A_2 πέτυχε (ενώ οι προηγούμενοι απέτυχαν).

Τέλος, ας θεωρήσουμε ότι υπάρχει μια βάση με δεδομένα για πτήσεις και ότι θέλουμε να κάνουμε επερωτήσεις με προτιμήσεις, χρησιμοποιώντας τους τελεστές προτίμησης.

Παράδειγμα 3.1. Θέλουμε ένα αεροπορικό εισιτήριο για Αθήνα και θα προτιμούσαμε θέση στο παράθυρο. Η πρόταση A αντιστοιχεί στο αν υπάρχει διαθέσιμη πτήση για Αθήνα και η πρόταση B αντιστοιχεί στο αν υπάρχει διαθέσιμη θέση στο παράθυρο. Η επερώτηση που θα κάνουμε στο σύστημα θα είναι η $A \wedge \mu B$.

Παράδειγμα 3.2. Θέλουμε ένα αεροπορικό εισιτήριο από Αθήνα για Λονδίνο από 27 Ιουλίου μέχρι 4 Αυγούστου. Αν δεν υπάρχουν εισιτήρια στις 4 Αυγούστου, θα θέλαμε να επιστρέψουμε στις 3 Αυγούστου ή εναλλακτικά στις 5 Αυγούστου. Επίσης, θα προτιμούσαμε απ'ευθείας πτήση. Έστω A : από Αθήνα, B : προς Λονδίνο, C : αναχώρηση 27 Ιουλίου, D : επιστροφή 4 Αυγούστου, E : επιστροφή 3 Αυγούστου, F : επιστροφή 5 Αυγούστου, G : απ'ευθείας πτήση. Η επερώτηση που θα κάνουμε στο σύστημα θα είναι η $A \wedge B \wedge C \wedge (D \vee \omega E \vee \omega^2 F) \wedge \mu G$.

Κεφάλαιο 4

Λογικός προγραμματισμός με προτιμήσεις

Στο κεφάλαιο αυτό θα ορίσουμε τον λογικό προγραμματισμό με προτιμήσεις, επεκτείνοντας τον κλασικό λογικό προγραμματισμό. Θα χρησιμοποιήσουμε τους τελεστές προτίμησης μ και ω στα σώματα των κανόνων, και θα εργαστούμε πάνω στην απειρότιμη λογική. Στη συνέχεια θα προτείνουμε μοντελοθεωρητική σημασιολογία σταθερού σημείου βασισμένη στα [46], [42].

4.1 Συντακτικό

Στην ενότητα αυτή θα ορίσουμε τα λογικά προγράμματα με προτιμήσεις, μια γενίκευση των κλασικών λογικών προγραμμάτων. Τα λογικά προγράμματα με προτιμήσεις αποτελούνται από κανόνες της μορφής $H \leftarrow B$, όπου B μια λογική πρόταση με προτιμήσεις, έτσι όπως ορίστηκε στο προηγούμενο κεφάλαιο.

Οι λογικές προτάσεις με προτιμήσεις μπορούν να παίρνουν τιμές από το άπειρο σύνολο αληθοτιμών $\mathbb{V} = \{F_0, F_1, \dots, 0, \dots, T_1, T_0\}$. Είναι επόμενο πως αυτή η διαβάθμιση στις τιμές των λογικών προτάσεων θα πρέπει να περάσει στο νόημα των κανόνων των λογικών προγραμμάτων με προτιμήσεις. Έτσι, το νόημα ενός κανόνα $H \leftarrow B$ είναι ότι η κεφαλή H είναι όσο αληθής όσο το σώμα B . Για παράδειγμα, ο κανόνας $A \leftarrow A_0 \wedge \mu A_1$ δηλώνει ότι η πρόταση A είναι όσο αληθής όσο η πρόταση « A_0 και προαιρετικά A_1 », ενώ ο κανόνας $A \leftarrow A_0 \vee \omega A_1$ δηλώνει ότι η πρόταση A είναι όσο αληθής όσο η πρόταση « A_0 ή εναλλακτικά A_1 ».

Στη συνέχεια θα ορίσουμε πιο τυπικά το συντακτικό των λογικών προγραμμάτων με προτιμήσεις.

Για κάθε λογικό πρόγραμμα με προτιμήσεις ορίζουμε τα εξής σύνολα:

- P : σύνολο κατηγορημάτων
- F : σύνολο συναρτησιακών συμβόλων
- V : σύνολο μεταβλητών

Σε κάθε κατηγορημα ή συναρτησιακό σύμβολο k αντιστοιχεί ένας ακέραιος αριθμός $n \geq 0$ που λέγεται βαθμός του k . Το k συμβολίζεται και k/n . Τα συναρτησιακά σύμβολα βαθμού 0 λέγονται και σταθερές.

Ορισμός 4.1 (Όρος). Ένας όρος ορίζεται ως εξής:

1. Αν $X \in V$, τότε το X είναι όρος.
2. Αν $c/0 \in F$, τότε το c είναι όρος.
3. Αν $f/n \in F$ και $t_1, t_2, \dots, t_n$ είναι όροι, τότε το $f(t_1, t_2, \dots, t_n)$ είναι όρος.

Ορισμός 4.2 (Ατομικός τύπος). Ένας ατομικός τύπος (ή άτομο) ορίζεται ως εξής:

1. Αν $p/0 \in P$, τότε το p είναι άτομο.
2. Αν $p/n \in P$ και $t_1, t_2, \dots, t_n$ είναι όροι, τότε το $p(t_1, t_2, \dots, t_n)$ είναι άτομο.

Ορισμός 4.3 (Λογικό πρόγραμμα με προτιμήσεις). Ένα λογικό πρόγραμμα με προτιμήσεις είναι ένα σύνολο κανόνων, καθένας από τους οποίους έχει μια από τις δύο μορφές:

$$A \leftarrow \mu^{i_1} A_1 \wedge \mu^{i_2} A_2 \wedge \dots \wedge \mu^{i_n} A_n, \quad n \geq 0, \quad i_k \geq 0$$

$$A \leftarrow \omega^{i_1} A_1 \vee \omega^{i_2} A_2 \vee \dots \vee \omega^{i_n} A_n, \quad n \geq 0, \quad i_k \geq 0$$

όπου τα $A, A_1, A_2, \dots, A_n$ είναι ατομικοί τύποι.

Παρατηρούμε ότι ο παραπάνω ορισμός αποτελεί γενίκευση του κλασικού λογικού προγραμματισμού, αν στα σώματα των κανόνων για κάθε k είναι $i_k = 0$.

Παράδειγμα 4.1.

$$P = \left\{ \begin{array}{l} s(X) \leftarrow \omega p(X), \\ p(X) \leftarrow q(X) \wedge \mu r(X), \\ q(a), r(a), q(b) \end{array} \right\}$$

Για τον όρο a , τα $s(a), p(a), q(a), r(a)$ είναι αληθή, αλλά το $s(a)$ θα είναι λιγότερο αληθές από το $p(a)$, αφού το $s(a)$ είναι τόσο αληθές όσο η πρόταση «εναλλακτικά $p(a)$ »,

ενώ το $p(a)$ θα είναι το ίδιο αληθές με το $r(a)$. Για τον όρο b , το $q(b)$ είναι αληθές και τα $s(b), p(b), r(b)$ είναι ψευδή, αλλά το $p(b)$ θα είναι λιγότερο ψευδές από το $r(b)$, αφού το $p(b)$ είναι τόσο αληθές όσο η πρόταση « $q(c)$ και προαιρετικά $r(c)$ », ενώ το $s(b)$ θα είναι το ίδιο ψευδές με το $p(b)$. Συνεπώς το νόημα του προγράμματος διαισθητικά θα μπορούσε να είναι το εξής:

	$s(\cdot)$	$p(\cdot)$	$q(\cdot)$	$r(\cdot)$
a	T_1	T_0	T_0	T_0
b	F_1	F_1	T_0	F_0

4.2 Μοντελοθεωρητική σημασιολογία

Στην ενότητα αυτή θα ορίσουμε τις έννοιες της απειρότιμης ερμηνείας και απειρότιμου μοντέλου για λογικά προγράμματα με προτιμήσεις. Όπως και στον κλασικό λογικό προγραμματισμό, και εδώ θα αγνοήσουμε το πραγματικό νόημα που κρύβεται πίσω από τις σταθερές και τα συναρτησιακά σύμβολα του προγράμματος.

Ορισμός 4.4 (Σύμπαν Herbrand). Έστω P λογικό πρόγραμμα με προτιμήσεις. Ορίζουμε ως σύμπαν Herbrand U_P το σύνολο όλων των όρων που δεν περιέχουν μεταβλητές και που μπορούν να παραχθούν από τις σταθερές και τα συναρτησιακά σύμβολα του P .

Ορισμός 4.5 (Βάση Herbrand). Έστω P λογικό πρόγραμμα με προτιμήσεις. Ορίζουμε ως βάση Herbrand B_P το σύνολο όλων των ατόμων που δεν περιέχουν μεταβλητές και που μπορούν να παραχθούν από τα κατηγορήματα του P με ορίσματα από το U_P .

Ορισμός 4.6 (Βασικό στιγμιότυπο). Ορίζουμε ως βασικό στιγμιότυπο $ground(P)$ ενός λογικού προγράμματος με προτιμήσεις P το (πιθανώς άπειρο) λογικό πρόγραμμα με προτιμήσεις το οποίο παράγεται αντικαθιστώντας στο P τις μεταβλητές με στοιχεία του U_P με όλους τους δυνατούς τρόπους.

Στον κλασικό λογικό προγραμματισμό, μια ερμηνεία ενός προγράμματος είναι ένα υποσύνολο I της βάσης Herbrand B_P . Αν κάποιο p ανήκει στο I , τότε στο p θα αντιστοιχίσουμε την τιμή T , διαφορετικά θα αντιστοιχίσουμε την τιμή F . Επειδή στην απειρότιμη λογική έχουμε άπειρες αληθοτιμές η ερμηνεία θα είναι μια συνάρτηση από άτομα της B_P στο σύνολο αληθοτιμών.

Ορισμός 4.7 (Απειρότιμη ερμηνεία). Ορίζουμε ως *απειρότιμη ερμηνεία* I ενός προγράμματος P μια συνάρτηση $I : B_P \rightarrow \mathbb{V}$, όπου B_P η βάση Herbrand του P και $\mathbb{V}$ το σύνολο απείρων τιμών αληθείας $\{0, F_0, T_0, F_1, T_1, \dots\}$.

Για να ορίσουμε την έννοια του απειρότιμου μοντέλου, θα χρειαστεί να επεκτείνουμε την έννοια της ερμηνείας και στους διαθέσιμους λογικούς τελεστές της γλώσσας μας $(\wedge, \vee, \mu, \omega)$ καθώς και στις σταθερές $True, False$.

Ορισμός 4.8. Έστω I απειρότιμη ερμηνεία ενός προγράμματος P , και $A, B \in B_P$. Η I μπορεί να επεκταθεί ως εξής:

- $I(A \wedge B) = \min\{I(A), I(B)\}$
- $I(A \vee B) = \max\{I(A), I(B)\}$
- $I(\mu^k A) = \begin{cases} T_i, & \text{αν } I(A) = T_i \\ F_{i+k}, & \text{αν } I(A) = F_i \\ 0, & \text{αν } I(A) = 0 \end{cases}$
- $I(\omega^k A) = \begin{cases} T_{i+k}, & \text{αν } I(A) = T_i \\ F_i, & \text{αν } I(A) = F_i \\ 0, & \text{αν } I(A) = 0 \end{cases}$
- $I(T) = T_0$

Ορισμός 4.9. Έστω I απειρότιμη ερμηνεία ενός προγράμματος P . Η I ικανοποιεί τον κανόνα $H \leftarrow \mathbf{B}$ όταν $I(H) \geq I(\mathbf{B})$.

Ορισμός 4.10. Μια απειρότιμη ερμηνεία I είναι μοντέλο όταν ικανοποιεί όλους τους κανόνες του P .

4.3 Σημασιολογία σταθερού σημείου

Στην ενότητα αυτή θα ορίσουμε τη σημασιολογία των λογικών προγραμμάτων με προτιμήσεις. Θα αποδείξουμε ότι υπάρχει ένα ελάχιστο απειρότιμο μοντέλο, παρουσιάζοντας ένα χαρακτηρισμό του μέσω σταθερού σημείου.

Η διαδικασία που θα ακολουθήσουμε είναι παρόμοια με αυτή που είδαμε στο Κεφάλαιο 2 για τον κλασικό λογικό προγραμματισμό. Θα χρησιμοποιήσουμε και εδώ κάποιους ορισμούς από τη θεωρία συνόλων καθώς και το θεώρημα του σταθερού

σημείου. Θα δείξουμε αρχικά ότι το σύνολο των απειρότιμων ερμηνειών ενός προγράμματος P , εφοδιασμένο με τη σχέση μερικής διάταξης $\leq$ είναι πλήρες πλέγμα. Στη συνέχεια θα ορίσουμε έναν τελεστή T_P , θα δείξουμε ότι είναι μονοτονικός και συνεχής, ώστε να είναι δυνατή η εφαρμογή του θεωρήματος του σταθερού σημείου.

4.3.1 Μερική διάταξη

Στο σημείο αυτό θα δείξουμε ότι το σύνολο των απειρότιμων ερμηνειών ενός προγράμματος P , εφοδιασμένο με τη σχέση μερικής διάταξης $\leq$ είναι πλήρες πλέγμα. Αρχικά θα ορίσουμε τη σχέση $\leq$ και θα δείξουμε ότι είναι σχέση μερικής διάταξης.

Ορισμός 4.11. Έστω I, J δύο απειρότιμες ερμηνείες ενός προγράμματος P . Θα γράφουμε $I \leq J$ αν για κάθε $p \in B_P$ ισχύει ότι $I(p) \leq J(p)$.

Πρόταση 4.1. Έστω $\mathcal{I}_P$ το σύνολο όλων των απειρότιμων ερμηνειών ενός προγράμματος P . Το σύνολο $\langle \mathcal{I}_P, \leq \rangle$ είναι μερικώς διατεταγμένο σύνολο.

Απόδειξη. Άρκεί να δείξουμε ότι η σχέση $\leq$ στο $\mathcal{I}_P$ είναι σχέση μερικής διάταξης. Θα χρησιμοποιήσουμε το ότι το σύνολο $\mathbb{V}$ των αληθοτιμών είναι εξ ορισμού ολικά διατεταγμένο. Έστω I, J, K απειρότιμες ερμηνείες του P .

- Για κάθε $A \in B_P : I(A) = I(A) \Rightarrow I(A) \leq I(A) \Rightarrow I \leq I$, άρα $\leq$ ανακλαστική.
- $I \leq J$ και $J \leq I \Rightarrow$ για κάθε $A \in B_P : I(A) \leq J(A)$ και $J(A) \leq I(A) \Rightarrow I(A) = J(A) \Rightarrow I = J$, άρα $\leq$ αντισυμμετρική.
- $I \leq J$ και $J \leq K \Rightarrow$ για κάθε $A \in B_P : I(A) \leq J(A)$ και $J(A) \leq K(A) \Rightarrow I(A) \leq K(A) \Rightarrow I \leq K$, άρα $\leq$ μεταβατική.

Η σχέση $\leq$ είναι ανακλαστική, αντισυμμετρική και μεταβατική, άρα είναι σχέση μερικής διάταξης. $\square$

Στη συνέχεια, θα δείξουμε ότι το σύνολο $\langle \mathcal{I}_P, \leq \rangle$ είναι πλήρες πλέγμα. Για το σκοπό αυτό, θα χρειαστούμε τα δύο παρακάτω λήμματα.

Λήμμα 4.1. Για κάθε σύνολο αληθοτιμών V ορίζονται τα $\text{lub}(V)$ και $\text{glb}(V)$.

Απόδειξη. Έστω $V_T = \{v \in V : v > 0\}$ και $V_F = \{v \in V : v < 0\}$. Αρχικά δείχνουμε ότι ορίζεται το $\text{lub}(V)$ και στη συνέχεια το $\text{glb}(V)$.

Αν $V_T \neq \emptyset$ τότε λόγω της διάταξης των αληθοτιμών θα υπάρχει σίγουρα κάποιο $v^* \in V_T$ τέτοιο ώστε $v^* \geq v$ για κάθε $v \in V_T$, άρα $\text{lub}(V) = v^*$. Αν $V_T = \emptyset$ τότε σίγουρα

το 0 είναι άνω φράγμα του V . Αν δεν υπάρχει κάποιο $v^* \in V_F$ τέτοιο ώστε $v^* \geq v$ για κάθε $v \in V_F$, τότε $\text{lub}(V) = 0$. Διαφορετικά, αν υπάρχει τέτοιο x^* , τότε $\text{lub}(V) = x^*$.

Αν $V_F \neq \emptyset$ τότε λόγω της διάταξης των αληθοτιμών θα υπάρχει σίγουρα κάποιο $v^* \in V_T$ τέτοιο ώστε $v^* \leq v$ για κάθε $v \in V_F$, άρα $\text{glb}(V) = x^*$. Αν $V_F = \emptyset$ τότε σίγουρα το 0 είναι κάτω φράγμα του V . Αν δεν υπάρχει κάποιο $v^* \in V_T$ τέτοιο ώστε $v^* \leq v$ για κάθε $v \in V_T$, τότε $\text{glb}(V) = 0$. Διαφορετικά, αν υπάρχει τέτοιο x^* , τότε $\text{glb}(V) = x^*$. $\square$

Λήμμα 4.2. Για κάθε σύνολο απειρότιμων ερμηνειών $\mathcal{X}$ ενός προγράμματος P ορίζονται τα $\text{lub}(\mathcal{X})$ και $\text{glb}(\mathcal{X})$.

Απόδειξη. Έστω $A \in B_P$. Για το σύνολο αληθοτιμών $\mathcal{X}(A) = \{I(A) : I \in \mathcal{X}\}$ ορίζονται τα $\text{lub}(\mathcal{X}(A))$ και $\text{glb}(\mathcal{X}(A))$, από το λήμμα 4.1. Άρα, αφού για κάθε $A \in B_P$ ορίζονται τα $\text{lub}(\mathcal{X}(A))$ και $\text{glb}(\mathcal{X}(A))$, συμπεραίνουμε ότι μπορούμε να ορίσουμε τα $\text{lub}(\mathcal{X}) \equiv \{(A, \text{lub}(\mathcal{X}(A))) : A \in B_P\}$ και $\text{glb}(\mathcal{X}) \equiv \{(A, \text{glb}(\mathcal{X}(A))) : A \in B_P\}$. $\square$

Πρόταση 4.2. Έστω $\mathcal{I}_P$ το σύνολο όλων των απειρότιμων ερμηνειών ενός προγράμματος P . Το σύνολο $\langle \mathcal{I}_P, \leq \rangle$ είναι πλήρες πλέγμα.

Απόδειξη. Από Λήμμα 2, για κάθε $\mathcal{X} \subseteq \mathcal{I}_P$ υπάρχουν τα $\text{glb}(\mathcal{X})$ και $\text{lub}(\mathcal{X})$. $\square$

4.3.2 Τελεστής T_P

Στο σημείο αυτό θα ορίσουμε τον τελεστή άμεσου επακόλουθου T_P . Ο τελεστής αυτός εφαρμόζεται πάνω στο σύνολο των απειρότιμων ερμηνειών ενός λογικού προγράμματος με προτιμήσεις, δηλαδή δέχεται μια ερμηνεία και μας δίνει μια νέα ερμηνεία. Με $T_P(I)(A)$ συμβολίζουμε την αληθοτιμή που αντιστοιχίζεται στο A σύμφωνα με την ερμηνεία $T_P(I)$.

Ορισμός 4.12. Έστω I απειρότιμη ερμηνεία και $\mathcal{I}_P$ το σύνολο όλων των ερμηνειών ενός προγράμματος P . Ορίζουμε τον τελεστή $T_P : \mathcal{I}_P \rightarrow \mathcal{I}_P$ ως εξής:

$$T_P(I)(A) = \text{lub}\{I(\mathbf{B}) : (A \leftarrow \mathbf{B}) \in \text{ground}(P)\}$$

4.3.3 Μονοτονικότητα του T_P

Στο σημείο αυτό θα αποδείξουμε ότι ο τελεστής T_P είναι $\leq$ -μονοτονικός. Η μονοτονικότητα του T_P θα αποδειχθεί σε στάδια, ξεκινώντας από τη μονοτονικότητα των τελεστών της γλώσσας.

Πρόταση 4.3. Οι τελεστές $\wedge, \vee$ είναι μονοτονικοί.

Απόδειξη. Έστω $X_1, X_2, Y_1, Y_2 \in \mathbb{V}$, με $X_1 \leq Y_1$ και $X_2 \leq Y_2$.

$$\begin{cases} X_1 \leq Y_1 \\ X_2 \leq Y_2 \end{cases} \implies \begin{cases} \max\{X_1, X_2\} \leq \max\{Y_1, Y_2\} \\ \min\{X_1, X_2\} \leq \min\{Y_1, Y_2\} \end{cases} \implies \begin{cases} X_1 \wedge X_2 \leq Y_1 \wedge Y_2 \\ X_1 \vee X_2 \leq Y_1 \vee Y_2 \end{cases}$$

□

Πρόταση 4.4. Οι τελεστές μ, ω είναι μονοτονικοί.

Απόδειξη. Έστω $X, Y \in \mathbb{V}$, με $X \leq Y$.

Αν $0 \leq X \leq Y$, τότε $\mu X = X$ και $\mu Y = Y$, και άρα $\mu X \leq \mu Y$. Αν $X < 0 \leq Y$, τότε $\mu X = \epsilon X$ και $\mu Y = Y$, και άρα $\mu X \leq \mu Y$. Αν $X \leq Y < 0$, τότε $\mu X = \epsilon X$ και $\mu Y = \epsilon Y$, και άρα $\mu X \leq \mu Y$.

Αν $0 \leq X \leq Y$, τότε $\omega X = \epsilon X$ και $\omega Y = \epsilon Y$, και άρα $\omega X \leq \omega Y$. Αν $X < 0 \leq Y$, τότε $\omega X = X$ και $\omega Y = \epsilon Y$, και άρα $\omega X \leq \omega Y$. Αν $X \leq Y < 0$, τότε $\omega X = X$ και $\omega Y = Y$, και άρα $\omega X \leq \omega Y$. □

Στη συνέχεια ακολουθεί ένα λήμμα στο οποίο δείχνουμε ότι τα σώματα των κανόνων είναι μονοτονικά, χρησιμοποιώντας τη μονοτονικότητα των τελεστών της γλώσσας. Αυτό το λήμμα θα χρειαστεί για την απόδειξη της μονοτονικότητας του T_P .

Λήμμα 4.3. Έστω I, J απειρότιμες ερμηνείες και $A \leftarrow \mathbf{B} \in \text{ground}(P)$. Αν $I \leq J$ τότε $I(\mathbf{B}) \leq J(\mathbf{B})$.

Απόδειξη. Θα δείξουμε το ζητούμενο για καθέναν από τους δύο τύπους κανόνων.

Έστω $\mathbf{B} = \bigwedge_{j=0}^n \mu^{i_j} A_j$. Επειδή οι τελεστές μ και $\wedge$ είναι μονοτονικοί, έχουμε:

$$\begin{aligned} I \leq J &\implies \begin{cases} I(A_0) \leq J(A_0) \\ I(A_1) \leq J(A_1) \\ \dots \\ I(A_n) \leq J(A_n) \end{cases} \implies \begin{cases} I(\mu^{i(0)} A_0) \leq J(\mu^{i(0)} A_0) \\ I(\mu^{i(1)} A_1) \leq J(\mu^{i(1)} A_1) \\ \dots \\ I(\mu^{i(n)} A_n) \leq J(\mu^{i(n)} A_n) \end{cases} \implies \\ &\implies I(\mu^{i(0)} A_0 \wedge \dots \wedge \mu^{i(n)} A_n) \leq J(\mu^{i(0)} A_0 \wedge \dots \wedge \mu^{i(n)} A_n) \implies I(\mathbf{B}) \leq J(\mathbf{B}). \end{aligned}$$

Έστω $\mathbf{B} = \bigvee_{j=0}^n \omega^{ij} A_j$. Επειδή οι τελεστές ω και $\vee$ είναι μονοτονικοί, έχουμε:

$$\begin{aligned} I \leq J &\implies \begin{cases} I(A_0) \leq J(A_0) \\ I(A_1) \leq J(A_1) \\ \dots \\ I(A_n) \leq J(A_n) \end{cases} \implies \begin{cases} I(\omega^{i(0)} A_0) \leq J(\omega^{i(0)} A_0) \\ I(\omega^{i(1)} A_1) \leq J(\omega^{i(1)} A_1) \\ \dots \\ I(\omega^{i(n)} A_n) \leq J(\omega^{i(n)} A_n) \end{cases} \implies \\ &\implies I(\omega^{i(0)} A_0 \vee \dots \vee \omega^{i(n)} A_n) \leq J(\omega^{i(0)} A_0 \vee \dots \vee \omega^{i(n)} A_n) \implies I(\mathbf{B}) \leq J(\mathbf{B}). \end{aligned}$$

□

Θεώρημα 4.1. Έστω πρόγραμμα P . Ο τελεστής T_P είναι μονοτονικός.

Απόδειξη. Έστω I, J απειρότιμες ερμηνείες του P , τέτοιες ώστε $I \leq J$. Αρκεί να δείξουμε ότι $T_P(I)(A) \leq T_P(J)(A)$ για κάθε $A \in B_P$.

Έστω $A \in B_P$. Χρησιμοποιώντας το λήμμα 4.3, και την πρόταση 1.1, έχουμε:

$$\begin{aligned} I \leq J &\implies I(\mathbf{B}) \leq J(\mathbf{B}), \text{ για κάθε } \mathbf{B} \in \{\mathbf{B} : A \leftarrow \mathbf{B} \in \text{ground}(P)\} \\ &\implies \text{lub}\{I(\mathbf{B}) : A \leftarrow \mathbf{B} \in \text{ground}(P)\} \leq \text{lub}\{J(\mathbf{B}) : A \leftarrow \mathbf{B} \in \text{ground}(P)\} \\ &\implies T_P(I)(A) \leq T_P(J)(A). \end{aligned}$$

□

4.3.4 Συνέχεια του T_P

Στο σημείο αυτό θα αποδείξουμε ότι ο τελεστής T_P είναι $\leq$ -συνεχής. Θα ξεκινήσουμε με τον ορισμό του συμβόλου $\nearrow$, που συμβολίζει ότι ένα άπειρο σύνολο αληθοτιμών αποτελείται από αληθοτιμές που διαισθητικά πλησιάζουν συνεχώς προς την «κεντρική» τιμή 0 χωρίς να τη φτάνουν ποτέ.

Ορισμός 4.13. Έστω S άπειρο σύνολο απειρότιμων αληθοτιμών. Θα γράφουμε $S \nearrow$ αν για κάθε $v \in S$ είναι $v < 0$ και για κάθε $v \in S$ υπάρχει $u \in S$ τέτοιο ώστε $v < u$.

Πρόταση 4.5. Έστω S άπειρο σύνολο αληθοτιμών. Είναι $S \nearrow \implies \text{lub}(S) = 0$.

Στη συνέχεια έχουμε δύο λήμματα, τα οποία διαισθητικά αποδεικνύουν την εξής συμπεριφορά: μας δίνεται ένα σύνολο ερμηνειών. Υπολογίζουμε το άνω όριο του (δεν

υπάρχει πρόβλημα, αφού το lub στις ερμηνείες ορίζεται πάντα), και αποτιμούμε το σώμα ενός κανόνα. Το αποτέλεσμα θα είναι μη μηδενική τιμή αν και μόνο αν υπάρχει κάποια ερμηνεία στο αρχικό σύνολο που αποτιμά το σώμα ακριβώς στην τιμή αυτή. Επίσης, το αποτέλεσμα θα είναι μηδέν αν και μόνο αν είτε υπάρχει κάποια ερμηνεία στο αρχικό σύνολο που αποτιμά το σώμα σε μηδέν, είτε όλες οι αποτιμήσεις στο αρχικό σύνολο πλησιάζουν συνεχώς προς το μηδέν χωρίς να το φτάνουν ποτέ. Σε κάθε περίπτωση, το lub των αποτιμήσεων όλων των ερμηνειών του συνόλου θα είναι ίσο με την αποτίμηση του lub όλων των ερμηνειών.

Λήμμα 4.4. Έστω $(A \leftarrow B) \in \text{ground}(P)$, και $\mathcal{X}$ κατευθυνόμενο σύνολο απειρότιμων ερμηνειών του P . Ισχύει ότι $\text{lub}(\mathcal{X})(B) = v, v \neq 0 \implies I^*(B) = v$ για κάποιο $I^* \in \mathcal{X}$.

Απόδειξη. Θα δείξουμε το ζητούμενο για καθέναν από τους δύο τύπους κανόνων.

Έστω $B = \bigwedge_{j=1}^n \mu^{ij} A_j$. Είναι

$$\text{lub}(\mathcal{X})(B) = v \implies \text{lub}(\mathcal{X})\left(\bigwedge_{j=1}^n \mu^{ij} A_j\right) = v \implies \bigwedge_{j=1}^n \mu^{ij} \text{lub}(\mathcal{X})(A_j) = v.$$

Για κάθε $1 \leq j \leq n$ υπάρχει $I_j \in \mathcal{X}$ τέτοιο ώστε $\bigwedge_{j=1}^n \mu^{ij} I_j(A_j) = v$. Έστω η ερμηνεία $I^* = \text{lub}\{I_j : 1 \leq j \leq n\}$. Επειδή το $\mathcal{X}$ είναι κατευθυνόμενο $I^* \in \mathcal{X}$. Αρκεί να δείξουμε ότι $I^*(B) = v$.

$$\bigwedge_{j=1}^n \mu^{ij} I_j(A_j) \leq \bigwedge_{j=1}^n \mu^{ij} I^*(A_j) \implies v \leq I\left(\bigwedge_{j=1}^n \mu^{ij} A_j\right) \implies I^*(B) \geq v.$$

$$I^* \leq \text{lub}(\mathcal{X}) \implies I^*(B) \leq \text{lub}(\mathcal{X})(B) \implies I^*(B) \leq v.$$

Έστω $B = \bigvee_{j=1}^n \omega^{ij} A_j$. Είναι

$$\text{lub}(\mathcal{X})(B) = v \implies \text{lub}(\mathcal{X})\left(\bigvee_{j=1}^n \omega^{ij} A_j\right) = v \implies \bigvee_{j=1}^n \omega^{ij} \text{lub}(\mathcal{X})(A_j) = v.$$

Για κάθε $1 \leq j \leq n$ υπάρχει $I_j \in \mathcal{X}$ τέτοιο ώστε $\bigvee_{j=1}^n \omega^{ij} I_j(A_j) = v$. Έστω η ερμηνεία $I^* = \text{lub}\{I_j : 1 \leq j \leq n\}$. Επειδή το $\mathcal{X}$ είναι κατευθυνόμενο $I^* \in \mathcal{X}$. Αρκεί να δείξουμε ότι $I^*(B) = v$.

$$\bigvee_{j=1}^n \omega^{ij} I_j(A_j) \leq \bigvee_{j=1}^n \omega^{ij} I^*(A_j) \implies v \leq I\left(\bigvee_{j=1}^n \omega^{ij} A_j\right) \implies I^*(B) \geq v.$$

$$I^* \leq \text{lub}(\mathcal{X}) \implies I^*(B) \leq \text{lub}(\mathcal{X})(B) \implies I^*(B) \leq v.$$

□

Λήμμα 4.5. Έστω $(A \leftarrow \mathbf{B}) \in \text{ground}(P)$, και $\mathcal{X}$ κατευθυνόμενο σύνολο απειρότιμων ερμηνειών του P . Ισχύει ότι $\text{lub}(\mathcal{X})(\mathbf{B}) = \text{lub}\{I(\mathbf{B}) : I \in \mathcal{X}\}$.

Απόδειξη. Λαμβάνουμε περιπτώσεις για το $\text{lub}(\mathcal{X})(\mathbf{B})$.

Αν $\text{lub}(\mathcal{X})(\mathbf{B}) = v$ με $v \neq 0$ τότε για κάθε $I \in \mathcal{X}$ είναι $I \leq \text{lub}(\mathcal{X}) \implies I(\mathbf{B}) \leq \text{lub}(\mathcal{X})(\mathbf{B}) \implies I(\mathbf{B}) \leq v$ και υπάρχει κάποιο $I^* \in \mathcal{X}$ τέτοιο ώστε $I^*(\mathbf{B}) = v$ (από Λήμμα 1). Άρα $\text{lub}\{I(\mathbf{B}) : I \in \mathcal{X}\} = v$.

Αν $\text{lub}(\mathcal{X})(\mathbf{B}) = 0$ τότε για κάθε $I \in \mathcal{X}$ είναι $I \leq \text{lub}(\mathcal{X}) \implies I(\mathbf{B}) \leq \text{lub}(\mathcal{X})(\mathbf{B}) \implies I(\mathbf{B}) \leq 0$ και:

- είτε υπάρχει κάποιο $I^* \in \mathcal{X}$ τέτοιο ώστε $I^*(\mathbf{B}) = 0$, και άρα $\text{lub}\{I(\mathbf{B}) : I \in \mathcal{X}\} = 0$.
- είτε δεν υπάρχει κάποιο $I^* \in \mathcal{X}$ τέτοιο ώστε $I^*(\mathbf{B}) = 0$ και $\{I(\mathbf{B}) : I \in \mathcal{X}\} \nearrow$ και άρα $\text{lub}\{I(\mathbf{B}) : I \in \mathcal{X}\} = 0$.

□

Θεώρημα 4.2. Έστω πρόγραμμα P . Ο τελεστής T_P είναι συνεχής.

Απόδειξη. Έστω $\mathcal{X}$ κατευθυνόμενο σύνολο απειρότιμων ερμηνειών του P . Άρκει να δείξουμε ότι $T_P(\text{lub}(\mathcal{X})) = \text{lub}\{T_P(I) : I \in \mathcal{X}\}$ ή ισοδύναμα ότι για κάθε $A \in B_P$ ισχύει ότι $T_P(\text{lub}(\mathcal{X}))(A) = \text{lub}\{T_P(I)(A) : I \in \mathcal{X}\}$.

$$\begin{aligned} T_P(\text{lub}(\mathcal{X}))(A) &= \text{lub}\{\text{lub}(\mathcal{X})(\mathbf{B}) : (A \leftarrow \mathbf{B}) \in \text{ground}(P)\} \\ &= \text{lub}\{\text{lub}\{I(\mathbf{B}) : I \in \mathcal{X}\} : (A \leftarrow \mathbf{B}) \in \text{ground}(P)\} \\ &= \text{lub}\{\text{lub}\{I(\mathbf{B}) : (A \leftarrow \mathbf{B}) \in \text{ground}(P)\} : I \in \mathcal{X}\} \\ &= \text{lub}\{T_P(I)(A) : I \in \mathcal{X}\} \end{aligned}$$

Η πρώτη και η τελευταία ισότητα είναι από τον ορισμό του T_P , η δεύτερη ισότητα από το λήμμα 4.5 και η τρίτη ισότητα από την πρόταση 1.2. □

Διαισθητικά, αν το $T_P(\text{lub}(\mathcal{X}))(A)$ ίσο με μια μη μηδενική τιμή v , θα υπάρχει κάποιος κανόνας με κεφαλή το A που θα αποτιμάται από το lub ακριβώς σε v και άρα θα υπάρχει μια ερμηνεία που αποτιμά το σώμα ακριβώς στην τιμή αυτή, και άρα το επαγόμενο της ερμηνείας αυτής θα είναι ίσο με v . Επειδή όλες οι άλλες ερμηνείες θα αποτιμούν το σώμα σε τιμές μικρότερες του v , τα αντίστοιχα επαγόμενα θα είναι μικρότερα του v , και άρα το lub των επαγόμενων των ερμηνειών του $\mathcal{X}$ θα είναι ακριβώς v . Στην περίπτωση όμως που έχουμε μηδενισμό, μπορεί να συμβαίνουν τα εξής:

- αν $T_P(\text{lub}(\mathcal{X}))(A) = 0$, τότε είτε κάποιο σώμα αποτιμάται από το lub σε 0 (οπότε είτε μια ερμηνεία αποτιμά το συγκεκριμένο σώμα σε 0 είτε όλες οι (άπειρες) ερμηνείες αποτιμούν το συγκεκριμένο σώμα σε τιμές που «πλησιάζουν συνεχώς» το 0), είτε το lub αποτιμά όλα τα (άπειρα) σώματα σε τιμές που «πλησιάζουν συνεχώς» το 0.
- αν $\text{lub}\{T_P(I)(A) : I \in \mathcal{X}\} = 0$, τότε είτε το επακόλουθο κάποιας ερμηνείας αποτιμά την κεφαλή σε 0 (οπότε είτε κάποιο σώμα αποτιμάται από τη συγκεκριμένη ερμηνεία σε 0 είτε όλα τα (άπειρα) σώματα αποτιμούνται από τη συγκεκριμένη ερμηνεία σε τιμές που «πλησιάζουν συνεχώς» το 0), είτε τα επακόλουθα όλων των (άπειρων) ερμηνειών αποτιμούν την κεφαλή σε τιμές που «πλησιάζουν συνεχώς» το 0.

Σύμφωνα με τα θεωρήματα σταθερού σημείου, επειδή ο τελεστής T_P είναι μονοτονικός και συνεχής, έχει ελάχιστο σταθερό σημείο. Τώρα μένει να δείξουμε ότι το σταθερό σημείο αυτό είναι και το ελάχιστο μοντέλο του προγράμματος P . Θα χρειαστούμε το επόμενο λήμμα.

Λήμμα 4.6. *Μια απειρότιμη ερμηνεία I είναι μοντέλο ενός προγράμματος P αν και μόνο αν $T_P(I) \leq I$.*

Απόδειξη. Η I είναι μοντέλο του $P \iff$ για κάθε $H \in B_P$ και για κάθε $H \leftarrow \mathbf{B} \in \text{ground}(P)$ είναι $I(H) \geq I(\mathbf{B}) \iff I(H) \geq \text{lub}\{I(\mathbf{B}) : (H \leftarrow \mathbf{B}) \in P\} \iff I(H) \geq T_P(I)(H) \iff T_P(I) \leq I. \quad \square$

Θεώρημα 4.3. *Έστω πρόγραμμα P . Είναι $M_P = T_P \uparrow \omega$.*

Απόδειξη. Χρησιμοποιώντας το λήμμα 4.6 και τα θεωρήματα 1.1 και 1.2, για το ελάχιστο μοντέλο M_P του P έχουμε:

$$M_P = \text{glb}\{I : I \text{ μοντέλο του } P\} = \text{glb}\{I \mid T_P(I) \leq I\} = \text{lfp}(T_P) = T_P \uparrow \omega.$$

$\square$

Το παραπάνω θεώρημα δηλώνει ότι το σταθερό σημείο και άρα το ελάχιστο μοντέλο μπορεί να προκύψει «κατασκευαστικά», δηλαδή εφαρμόζοντας τον T_P στον εαυτό του πολλές φορές, δηλαδή:

$$\perp, T_P(\perp), T_P(T_P(\perp)), T_P(T_P(T_P(\perp))), \dots$$

Το στοιχείο $\perp$ θα είναι το glb του συνόλου των ερμηνειών του προγράμματος P , δηλαδή η ερμηνεία $\{(p, F_0) : p \in B_P\}$. Ας δουμε ένα παράδειγμα:

Παράδειγμα 4.2. Έστω το πρόγραμμα:

$$P = \left\{ \begin{array}{l} s \leftarrow \omega p. \\ s \leftarrow q \wedge \mu r. \\ p \leftarrow q. \\ q. \end{array} \right\}$$

Οι διαδοχικές εφαρμογές του T_P είναι οι εξής:

- $T_P \uparrow 0 = \{(s, F_0), (p, F_0), (q, F_0), (r, F_0)\}$
- $T_P \uparrow 1 = \{(s, F_0), (p, F_0), (q, \mathbf{T}_0), (r, F_0)\}$
- $T_P \uparrow 2 = \{(s, \mathbf{F}_1), (p, T_0), (q, T_0), (r, F_0)\}$
- $T_P \uparrow 3 = \{(s, \mathbf{T}_1), (p, T_0), (q, T_0), (r, F_0)\}$
- $T_P \uparrow 4 = \{(s, T_1), (p, T_0), (q, T_0), (r, F_0)\}$

Παράδειγμα 4.3. Έστω το πρόγραμμα:

$$P = \left\{ \begin{array}{l} s(X) \leftarrow \omega p(X), \\ p(X) \leftarrow q(X) \wedge \mu r(X), \\ q(a), r(a), q(b) \end{array} \right\}$$

Το βασικό στιγμιότυπο του P είναι:

$$\text{ground}(P) = \left\{ \begin{array}{l} s(a) \leftarrow \omega p(a), \quad p(a) \leftarrow q(a) \wedge \mu r(a), \\ s(b) \leftarrow \omega p(b), \quad p(b) \leftarrow q(b) \wedge \mu r(b), \\ q(a), r(a), q(b) \end{array} \right\}$$

Οι διαδοχικές εφαρμογές του T_P είναι οι εξής:

- $T_P \uparrow 0 = \left\{ \begin{array}{l} (s(a), F_0), \quad (p(a), F_0), \quad (q(a), F_0), \quad (r(a), F_0), \\ (s(b), F_0), \quad (p(b), F_0), \quad (q(b), F_0), \quad (r(b), F_0), \end{array} \right\}$
- $T_P \uparrow 1 = \left\{ \begin{array}{l} (s(a), F_0), \quad (p(a), F_0), \quad (q(a), \mathbf{T}_0), \quad (r(a), \mathbf{T}_0), \\ (s(b), F_0), \quad (p(b), F_0), \quad (q(b), \mathbf{T}_0), \quad (r(b), F_0), \end{array} \right\}$

$$\begin{aligned}
 \bullet \quad T_P \uparrow 2 &= \left\{ \begin{array}{cccc} (s(a), F_0), & (p(a), \mathbf{T}_0), & (q(a), T_0), & (r(a), T_0), \\ (s(b), F_0), & (p(b), \mathbf{F}_1), & (q(b), T_0), & (r(b), F_0), \end{array} \right\} \\
 \bullet \quad T_P \uparrow 3 &= \left\{ \begin{array}{cccc} (s(a), \mathbf{T}_1), & (p(a), T_0), & (q(a), T_0), & (r(a), T_0), \\ (s(b), \mathbf{F}_1), & (p(b), F_1), & (q(b), T_0), & (r(b), F_0), \end{array} \right\} \\
 \bullet \quad T_P \uparrow 4 &= \left\{ \begin{array}{cccc} (s(a), T_1), & (p(a), T_0), & (q(a), T_0), & (r(a), T_0), \\ (s(b), F_1), & (p(b), F_1), & (q(b), T_0), & (r(b), F_0), \end{array} \right\}
 \end{aligned}$$

4.4 Τομή μοντέλων

Στην ενότητα αυτή θα παρουσιάσουμε έναν εναλλακτικό τρόπο χαρακτηρισμού του ελαχίστου μοντέλου ενός λογικού προγράμματος με προτιμήσεις P . Ο χαρακτηρισμός αυτός είναι μια γενίκευση του γνωστού θεωρήματος τομής μοντέλων για τα κλασικά λογικά προγράμματα.

Θεώρημα 4.4. Έστω $\mathcal{M} = \{M_1, M_2, \dots\}$ ένα σύνολο απειρότιμων μοντέλων ενός προγράμματος P . Τότε και η ερμηνεία $glb(\mathcal{M})$ είναι μοντέλο του P .

Απόδειξη. Έστω $glb(\mathcal{M})$ όχι μοντέλο του P . Άρα για κάποιο κανόνα $(A \leftarrow \mathbf{B}) \in ground(P)$ θα ισχύει $glb(\mathcal{M})(A) < glb(\mathcal{M})(\mathbf{B})$. Θα καταλήξουμε σε άτοπο για καθέναν από τους δύο τύπους κανόνων του P .

Έστω $\mathbf{B} = \bigwedge_{j=1}^n \mu^{ij} A_j$. Επίσης έστω $glb(\mathcal{M})(A) = v$ και $glb(\mathcal{M})(\mu^{ij} A_j) = v_j$ για κάθε $1 \leq j \leq n$. Αφού ισχύει ότι

$$glb(\mathcal{M})(A) < glb(\mathcal{M})\left(\bigwedge_{j=1}^n \mu^{ij} A_j\right) = \min \left\{ glb(\mathcal{M})(\mu^{ij} A_j) : 1 \leq j \leq n \right\},$$

θα ισχύει και $v < v_j$, για κάθε $1 \leq j \leq n$. Επίσης,

$$M \geq glb(\mathcal{M}) \implies M(\mu^{ij} A_j) \geq glb(\mathcal{M})(\mu^{ij} A_j) = v_j,$$

για κάθε $1 \leq j \leq n$ και για κάθε $M \in \mathcal{M}$. Όμως κάθε $M \in \mathcal{M}$ είναι μοντέλο του P άρα:

$$\begin{aligned}
 M(A) &\geq M\left(\bigwedge_{j=1}^n \mu^{ij} A_j\right) = \min\{M(\mu^{ij} A_j) : 1 \leq j \leq n\} \\
 &\geq M(\mu^{i_{k_M}} A_{k_M}) \geq v_{k_M}, \text{ για κάποιο } 1 \leq k_M \leq n
 \end{aligned}$$

Συνεπώς $glb\{M(A) : M \in \mathcal{M}\} \geq glb\{v_{k(M)} : M \in \mathcal{M}\} > v \implies glb(\mathcal{M})(A) > v$. Άτοπο, γιατί $glb(\mathcal{M})(A) = v$.

Έστω $\mathbf{B} = \bigvee_{j=1}^n \omega^{i_j} A_j$. Επίσης έστω $glb(\mathcal{M})(A) = v$ και $glb(\mathcal{M})(\omega^{i_j} A_j) = v_j$ για κάθε $1 \leq j \leq n$. Αφού ισχύει ότι

$$glb(\mathcal{M})(A) < glb(\mathcal{M})\left(\bigvee_{j=1}^n \omega^{i_j} A_j\right) = \max\left\{glb(\mathcal{M})(\omega^{i_j} A_j) : 1 \leq j \leq n\right\},$$

θα ισχύει και $v < v_k$, για κάποιο $1 \leq k \leq n$. Επίσης,

$$M \geq glb(\mathcal{M}) \implies M(\omega^{i_j} A_j) \geq glb(\mathcal{M})(\omega^{i_j} A_j) = v_j,$$

για κάθε $1 \leq j \leq n$ και για κάθε $M \in \mathcal{M}$. Όμως κάθε $M \in \mathcal{M}$ είναι μοντέλο του P άρα:

$$\begin{aligned} M(A) &\geq M\left(\bigvee_{j=1}^n \omega^{i_j} A_j\right) = \max\{M(\omega^{i_j} A_j) : 1 \leq j \leq n\} \\ &\geq M(\omega^{i_k} A_k) \geq v_k, \text{ για καθε } 1 \leq k \leq n \end{aligned}$$

Συνεπώς $glb\{M(A) : M \in \mathcal{M}\} \geq glb\{v_j : 1 \leq j \leq n\} \implies glb(\mathcal{M})(A) \geq v_j$, για κάθε $1 \leq j \leq n$. Άτοπο, γιατί $glb(\mathcal{M})(A) < v_k$. $\square$

Παράδειγμα 4.4. Έστω το πρόγραμμα:

$$P = \left\{ \begin{array}{l} s(X) \leftarrow \omega p(X), \\ p(X) \leftarrow q(X) \wedge \mu r(X), \\ q(a), r(a), q(b) \end{array} \right\}$$

και οι ερμηνείες:

$$M_1 = \left\{ \begin{array}{llll} (s(a), T_0), & (p(a), T_0), & (q(a), T_0), & (r(a), T_0), \\ (s(b), F_2), & (p(b), F_1), & (q(b), T_0), & (r(b), F_0), \end{array} \right\}$$

$$M_2 = \left\{ \begin{array}{llll} (s(a), T_1), & (p(a), T_0), & (q(a), T_0), & (r(a), T_0), \\ (s(b), F_2), & (p(b), F_2), & (q(b), T_0), & (r(b), F_1), \end{array} \right\}$$

$$glb\{M_1, M_2\} = \left\{ \begin{array}{llll} (s(a), T_1), & (p(a), T_0), & (q(a), T_0), & (r(a), T_0), \\ (s(b), F_2), & (p(b), F_1), & (q(b), T_0), & (r(b), F_0), \end{array} \right\}$$

Μπορεί κανείς να επιβεβαιώσει ότι τα $M_1, M_2, glb\{M_1, M_2\}$ είναι μοντέλα του P .

Καθώς για κάθε πρόγραμμα P η ερμηνεία $\{(p, T_0) : p \in B_P\}$ είναι μοντέλο του P , άρα το σύνολο των μοντέλων του P είναι μη κενό. Συνεπώς το glb όλων των μοντέλων του P είναι το ελάχιστο μοντέλο του P .

Η παραπάνω ιδιότητα είναι σημαντική αλλά δεν προσφέρεται για πρακτικούς υπολογισμούς, αφού ακόμα και για μικρά πεπερασμένα προγράμματα μηδενικής τάξης μπορεί να έχουμε άπειρα μοντέλα. Για παράδειγμα το πρόγραμμα $P = \{p \leftarrow \mu q\}$ έχει το εξής σύνολο μοντέλων:

$$\mathcal{M}_P = \{\{(p, F_i), (q, F_j)\} : i > j\} \cup \{\{(p, T_i), (q, F_j)\}\} \cup \{\{(p, T_i), (q, T_j)\} : i \geq j\}$$

Προφανώς, $M_P = glb(\mathcal{M}_P) = \{(p, F_1), (q, F_0)\}$.

Κεφάλαιο 5

Εφαρμογές

Στο κεφάλαιο αυτό θα παρουσιάσουμε κάποιες εφαρμογές ή παραδείγματα λογικών προγραμμάτων με προτιμήσεις. Για κάθε πρόγραμμα αρχικά θα περιγράψουμε διαισθητικά το νόημα του και το προσδωκόμενο αποτέλεσμα που θα θέλαμε να μας επιστρέψει το σύστημα προτιμήσεων. Στη συνέχεια, θα βρούμε το ελάχιστο απειρότιμο μοντέλο του προγράμματος με διαδοχικές εφαρμογές του τελεστή T_P , έτσι όπως ορίστηκε στο προηγούμενο κεφάλαιο.

5.1 Προτιμήσεις σε ξενοδοχεία

Στον παρακάτω πίνακα περιλαμβάνεται μια λίστα με ξενοδοχεία. Για κάθε ξενοδοχείο υπάρχει η διαθεσιμότητα καθώς και οι συμπεριλαμβανόμενες παροχές.

	h_1	h_2	h_3	h_4	h_5	h_6	h_7	h_8
διαθεσιμότητα	✓	✓	✓	✓	✓	✓		
κλιματισμός	✓			✓	✓		✓	
πρωινό		✓		✓		✓	✓	✓
Wi-fi			✓		✓	✓	✓	

Δύο φίλοι θέλουν να κλείσουν ένα δωμάτιο από τα ξενοδοχεία αυτά. Εκτός από το (προφανές) ότι πρέπει να υπάρχουν διαθέσιμα δωμάτια, ο καθένας από τους δύο έχει τις δικές του προτιμήσεις ανάλογα με τις παροχές. Για παράδειγμα:

- Ο πρώτος απαιτεί να υπάρχουν διαθέσιμα δωμάτια, ενώ θα ήθελε προαιρετικά να υπάρχει σύνδεση wifi και πιο προαιρετικά να προσφέρεται πρωινό.

- Ο δεύτερος απαιτεί να υπάρχουν διαθέσιμα δωμάτια, ενώ θα ήθελε προαιρετικά να προσφέρεται πρωινό και πιο προαιρετικά να υπάρχει κλιματισμός.

Τέλος, οι δυο φίλοι θέλουν να συνεννοηθούν κατάλληλα ώστε να επιλέξουν το ξενοδοχείο έτσι ώστε να είναι και οι δυο όσο το δυνατόν περισσότερο ικανοποιημένοι.

Στη συνέχεια θα περιγράψουμε τον τρόπο κατασκευής του λογικού προγράμματος με προτιμήσεις για το πρόβλημα αυτό.

Αρχικά, για κάθε ξενοδοχείο ορίζουμε τα κατηγορήματα $avail/1$, $ac/1$, $bf/1$, $wifi/1$, τα οποία αντιστοιχούν στη διαθεσιμότητα, την ύπαρξη κλιματισμού, την προσφορά πρωινού και τη σύνδεση wifi. Στη συνέχεια έχουμε τους ορισμούς των προτιμήσεων των χρηστών. Για κάθε χρήστη και για κάθε ξενοδοχείο ορίζουμε το κατηγορήμα $pref/2$ που αντιστοιχεί στη γνώμη του χρήστη για το ξενοδοχείο ανάλογα με τα κριτήρια προτίμησης που έχει θέσει. Τέλος, για κάθε ξενοδοχείο ορίζουμε το κατηγορήμα $mark/1$ που αντιστοιχεί στη γνώμη όλων των χρηστών για το ξενοδοχείο ανάλογα με τα κριτήρια προτίμησης που έχουν θέσει. Το πρόγραμμα για αυτό το πρόβλημα είναι το εξής:

$$avail(h_1). \quad avail(h_2). \quad avail(h_3). \quad avail(h_4). \quad avail(h_5). \quad avail(h_6).$$

$$ac(h_1). \quad ac(h_4). \quad ac(h_5). \quad ac(h_7).$$

$$bf(h_2). \quad bf(h_4). \quad bf(h_6). \quad bf(h_7). \quad bf(h_8).$$

$$wifi(h_3). \quad wifi(h_5). \quad wifi(h_6). \quad wifi(h_7).$$

$$pref(a, X) \leftarrow avail(X) \wedge \mu wifi(X) \wedge \mu^2 bf(X).$$

$$pref(b, X) \leftarrow avail(X) \wedge \mu bf(X) \wedge \mu^2 ac(X).$$

$$mark(X) \leftarrow pref(a, X) \wedge pref(b, X).$$

Στη συνέχεια θα εξετάσουμε διαισθητικά τι τιμή αληθείας περιμένουμε να πάρουμε ως αποτέλεσμα για το κατηγορήμα $mark$.

- Τα άτομα $mark(h_1), mark(h_3), mark(h_5)$ θα πάρουν την τιμή F_1 , γιατί δεν έχουν πρωινό, που είναι προτίμηση πρώτου βαθμού για τον b . Παρόλο που τα h_3, h_5 έχουν περισσότερες παροχές, και τα τρία άτομα έχουν την ίδια αληθοτιμή επειδή το επίπεδο 1 είναι απείρως σημαντικότερο από το επίπεδο 2.
- Τα άτομα $mark(h_2), mark(h_4)$ θα πάρουν την τιμή F_1 , γιατί δεν έχουν wifi, που είναι προτίμηση πρώτου βαθμού για τον a . Παρόλο που τα h_4 έχει περισσότερες παροχές, και τα δύο άτομα έχουν την ίδια αληθοτιμή επειδή το επίπεδο 1 είναι απείρως σημαντικότερο από το επίπεδο 2.

- Το άτομο $mark(h_6)$ θα πάρει την τιμή F_2 , γιατί δεν έχει κλιματισμό, που είναι προτίμηση δεύτερου βαθμού για τον b .
- Τα άτομα $mark(h_7), mark(h_8)$ θα πάρουν την τιμή F_0 , γιατί δεν έχουν διαθέσιμα δωμάτια.

Προφανώς, το προτιμότερο ξενοδοχείο είναι αυτό με τη μεγαλύτερη αληθοτιμή, δηλαδή το ξενοδοχείο h_6 . Ο a είναι πλήρως ευχαριστημένος γιατί έχει και σύνδεση wifi και πρωινό, ενώ η απουσία του κλιματισμού δεν «ενοχλεί» τόσο τον b , όσο θα τον ενοχλούσε π.χ. η απουσία πρωινού.

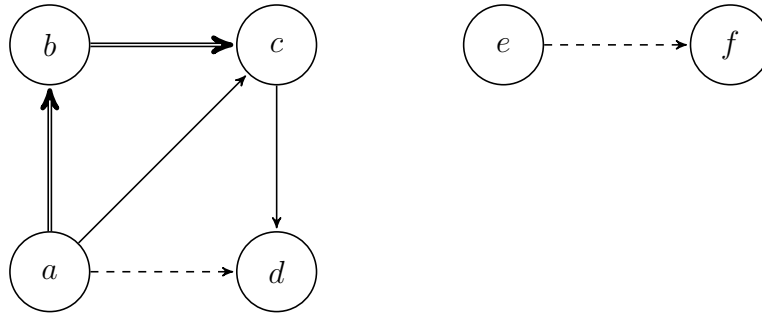
Οι διαδοχικές εφαρμογές του T_P είναι οι εξής (όποια άτομα δεν αναφέρονται θεωρείται ότι παίρνουν την τιμή F_0):

1. $(avail(h_1), \mathbf{T}_0), (avail(h_2), \mathbf{T}_0), (avail(h_3), \mathbf{T}_0), (avail(h_4), \mathbf{T}_0), (avail(h_5), \mathbf{T}_0), (avail(h_6), \mathbf{T}_0), (ac(h_1), \mathbf{T}_0), (ac(h_4), \mathbf{T}_0), (ac(h_5), \mathbf{T}_0), (ac(h_7), \mathbf{T}_0), (bf(h_2), \mathbf{T}_0), (bf(h_4), \mathbf{T}_0), (bf(h_6), \mathbf{T}_0), (bf(h_7), \mathbf{T}_0), (bf(h_8), \mathbf{T}_0), (wifi(h_3), \mathbf{T}_0), (wifi(h_5), \mathbf{T}_0), (wifi(h_6), \mathbf{T}_0), (wifi(h_7), \mathbf{T}_0).$
2. $(avail(h_1), T_0), (avail(h_2), T_0), (avail(h_3), T_0), (avail(h_4), T_0), (avail(h_5), T_0), (avail(h_6), T_0), (ac(h_1), T_0), (ac(h_4), T_0), (ac(h_5), T_0), (ac(h_7), T_0), (bf(h_2), T_0), (bf(h_4), T_0), (bf(h_6), T_0), (bf(h_7), T_0), (bf(h_8), T_0), (wifi(h_3), T_0), (wifi(h_5), T_0), (wifi(h_6), T_0), (wifi(h_7), T_0), (pref(a, h_1), \mathbf{F}_1), (pref(a, h_2), \mathbf{F}_1), (pref(a, h_3), \mathbf{F}_2), (pref(a, h_4), \mathbf{F}_1), (pref(a, h_5), \mathbf{F}_2), (pref(a, h_6), \mathbf{T}_0), (pref(b, h_1), \mathbf{F}_1), (pref(b, h_2), \mathbf{F}_2), (pref(b, h_3), \mathbf{F}_1), (pref(b, h_4), \mathbf{T}_0), (pref(b, h_5), \mathbf{F}_1), (pref(b, h_6), \mathbf{F}_2).$
3. $(avail(h_1), T_0), (avail(h_2), T_0), (avail(h_3), T_0), (avail(h_4), T_0), (avail(h_5), T_0), (avail(h_6), T_0), (ac(h_1), T_0), (ac(h_4), T_0), (ac(h_5), T_0), (ac(h_7), T_0), (bf(h_2), T_0), (bf(h_4), T_0), (bf(h_6), T_0), (bf(h_7), T_0), (bf(h_8), T_0), (wifi(h_3), T_0), (wifi(h_5), T_0), (wifi(h_6), T_0), (wifi(h_7), T_0), (pref(a, h_1), F_1), (pref(a, h_2), F_1), (pref(a, h_3), F_2), (pref(a, h_4), F_1), (pref(a, h_5), F_2), (pref(a, h_6), T_0), (pref(b, h_1), F_1), (pref(b, h_2), F_2), (pref(b, h_3), F_1), (pref(b, h_4), T_0), (pref(b, h_5), F_1), (pref(b, h_6), F_2), (mark(h_1), \mathbf{F}_1), (mark(h_2), \mathbf{F}_1), (mark(h_3), \mathbf{F}_1), (mark(h_4), \mathbf{F}_1), (mark(h_5), \mathbf{F}_1), (mark(h_6), \mathbf{F}_2).$
4. $(avail(h_1), T_0), (avail(h_2), T_0), (avail(h_3), T_0), (avail(h_4), T_0), (avail(h_5), T_0), (avail(h_6), T_0), (ac(h_1), T_0), (ac(h_4), T_0), (ac(h_5), T_0), (ac(h_7), T_0), (bf(h_2), T_0), (bf(h_4), T_0), (bf(h_6), T_0), (bf(h_7), T_0), (bf(h_8), T_0), (wifi(h_3), T_0), (wifi(h_5), T_0), (wifi(h_6), T_0), (wifi(h_7), T_0), (pref(a, h_1), F_1), (pref(a, h_2), F_1), (pref(a, h_3), F_2), (pref(a, h_4), F_1), (pref(a, h_5), F_2), (pref(a, h_6), T_0), (pref(b, h_1), F_1), (pref(b, h_2), F_2), (pref(b, h_3), F_1), (pref(b, h_4), T_0), (pref(b, h_5), F_1), (pref(b, h_6), F_2),$

$(\text{mark}(h_1), F_1), (\text{mark}(h_2), F_1), (\text{mark}(h_3), F_1), (\text{mark}(h_4), F_1), (\text{mark}(h_5), F_1),$
 $(\text{mark}(h_6), F_2).$

5.2 Προτιμήσεις στο οδικό δίκτυο

Στο παρακάτω γράφημα εικονίζεται ένα γράφημα-χάρτης, στο οποίο ένας κόμβος είναι ένα χωριό και μια ακμή είναι ένας δρόμος που συνδέει τα δυο χωριά. Ένας δρόμος μπορεί να είναι είτε φαρδύς ασφάλτινος δρόμος (συμβολίζεται με διπλή γραμμή), είτε στενός ασφάλτινος δρόμος (συμβολίζεται με απλή γραμμή), είτε χωματόδρομος (συμβολίζεται με διακεκομένη γραμμή).



Ένας οδηγός που θέλει να περιηγηθεί σε αυτό το δίκτυο ιεραρχεί τις προτιμήσεις του ως προς το είδος του δρόμου ως εξής: προτιμάει έναν φαρδύ ασφάλτινο δρόμο από έναν στενό, και έναν στενό ασφάλτινο δρόμο από έναν χωματόδρομο. Ο οδηγός θέλει αρχικά να εξετάσει αν υπάρχει δυνατότητα να μεταβεί από ένα χωριό στο άλλο, και σε περίπτωση που η μετάβαση είναι δυνατή, να γνωρίζει κατά πόσον είναι αναγκασμένος να περάσει από δρόμο χαμηλότερης προτίμησης. Για παράδειγμα, στο προηγούμενο γράφημα για τη μετάβαση από το a στο c υπάρχει η δυνατότητα μετάβασης χρησιμοποιώντας μόνο φαρδύ ασφάλτινο δρόμο (μέσω του b), ενώ για τη μετάβαση από το a στο d είναι αναγκασμένος να περάσει από τον στενό δρόμο cd . Τα e, f είναι απροσπέλαστα από τον a (ίσως να βρίσκονται σε κάποιο νησί).

Στη συνέχεια θα περιγράψουμε τον τρόπο κατασκευής του λογικού προγράμματος με προτιμήσεις για το πρόβλημα αυτό.

Στο πρώτο τμήμα του προγράμματος έχουμε τον ορισμό του χάρτη. Για κάθε δρόμο του χάρτη ορίζουμε το κατηγορημα $\text{edge}/3$, με τα δύο πρώτα ορίσματα να είναι τα χωριά που συνδέονται και το τρίτο να είναι ο τύπος του δρόμου. Θα χρησιμοποιήσουμε τα σύμβολα $\implies, \longrightarrow, \dashrightarrow$ που θα αντιστοιχούν στον φαρδύ ασφάλτινο, στενό ασφάλτινο και χωματόδρομο αντίστοιχα.

Στη συνέχεια έχουμε τους ορισμούς των προτιμήσεων του χρήστη. Ορίζουμε το κατηγορημα $mark/2$ το οποίο θα παίρνει την τιμή αληθείας που αντιστοιχεί στην προτίμηση του χρήστη ανάλογα με τον τύπο της ακμής. Διαισθητικά, το $mark$ είναι ένα «ποσοτικό» $edge$ με τιμή αληθείας ανάλογη με τις προτιμήσεις του οδηγού. Στο παραδειγμά μας, αν ο φαρδύς ασφάλτινος δρόμος αντιστοιχεί στην τιμή T_0 , ο στενός ασφάλτινος δρόμος αντιστοιχεί στην τιμή T_1 και ο χωματόδρομος στην τιμή T_2 , και η μη ύπαρξη δρόμου στην τιμή F_0 .

Τέλος, έχουμε τον τελικό αναδρομικό ορισμό $reach/2$ που είναι ένα κλασικό πρόγραμμα εύρεσης μονοπατιού σε γράφους, το οποίο όμως δεν χρησιμοποιεί το $edge$, αλλά το $mark$. Συνεπώς το τελικό πρόγραμμα για το πρόβλημα αυτό είναι το εξής:

$$\begin{aligned} & edge(a, b, \implies). \quad edge(a, c, \longrightarrow). \quad edge(a, d, \dashrightarrow). \\ & edge(b, c, \implies). \quad edge(c, d, \longrightarrow). \quad edge(e, f, \dashrightarrow). \\ & mark(X, Y) \leftarrow edge(X, Y, \implies) \vee \omega \, edge(X, Y, \longrightarrow) \vee \omega^2 \, edge(X, Y, \dashrightarrow). \\ & reach(X, Y) \leftarrow mark(X, Y). \\ & reach(X, Y) \leftarrow reach(X, Z) \wedge mark(Z, Y). \end{aligned}$$

Στη συνέχεια θα εξετάσουμε διαισθητικά τι τιμή αληθείας περιμένουμε να πάρουμε ως αποτέλεσμα για το κατηγορημα $reach$.

- Το άτομο $reach(a, b)$ θα πάρει την τιμή T_0 , λόγω του δρόμου ab .
- Το άτομο $reach(a, c)$ θα πάρει την τιμή T_0 , γιατί μπορούμε να μην πάμε κατευθείαν από τον στενό δρόμο ac , αλλά να πάμε μέσω του b .
- Το άτομο $reach(a, d)$ θα πάρει την τιμή T_1 , γιατί παρόλο που μπορούμε να μην χρησιμοποιήσουμε τον χωματόδρομο ad , πηγαίνοντας μέσω του b δεν μπορούμε να αποφύγουμε τον στενό δρόμο cd .
- Το άτομο $reach(b, c)$ θα πάρει την τιμή T_0 , λόγω του δρόμου bc .
- Το άτομο $reach(b, d)$ θα πάρει την τιμή T_1 , γιατί δεν μπορούμε να αποφύγουμε τον στενό δρόμο cd .
- Το άτομο $reach(c, d)$ θα πάρει την τιμή T_1 , λόγω του δρόμου cd που είναι και ο μοναδικός μεταξύ τους.
- Το άτομο $reach(e, f)$ θα πάρει την τιμή T_2 , λόγω του δρόμου ef που είναι και ο μοναδικός μεταξύ τους.
- Τα άτομα $reach(a, e)$, $reach(a, f)$, $reach(b, e)$, $reach(b, f)$, $reach(c, e)$, $reach(c, f)$, $reach(d, e)$, $reach(d, f)$ θα πάρουν την τιμή F_0 γιατί δεν υπάρχει μονοπάτι που

να συνδέει τα αντίστοιχα χωριά στον χάρτη.

Οι διαδοχικές εφαρμογές του T_P είναι οι εξής (όποια άτομα δεν αναφέρονται θεωρείται ότι παίρνουν την τιμή F_0):

1. $(edge(a, b, \implies), \mathbf{T}_0), (edge(a, c, \longrightarrow), \mathbf{T}_0), (edge(a, d, \dashrightarrow), \mathbf{T}_0), (edge(b, c, \implies), \mathbf{T}_0), (edge(c, d, \longrightarrow), \mathbf{T}_0), (edge(e, f, \dashrightarrow), \mathbf{T}_0).$
2. $(edge(a, b, \implies), T_0), (edge(a, c, \longrightarrow), T_0), (edge(a, d, \dashrightarrow), T_0), (edge(b, c, \implies), T_0), (edge(c, d, \longrightarrow), T_0), (edge(e, f, \dashrightarrow), T_0), (mark(a, b), \mathbf{T}_0), (mark(a, c), \mathbf{T}_1), (mark(a, d), \mathbf{T}_2), (mark(b, c), \mathbf{T}_0), (mark(c, d), \mathbf{T}_1), (mark(e, f), \mathbf{T}_2).$
3. $(edge(a, b, \implies), T_0), (edge(a, c, \longrightarrow), T_0), (edge(a, d, \dashrightarrow), T_0), (edge(b, c, \implies), T_0), (edge(c, d, \longrightarrow), T_0), (edge(e, f, \dashrightarrow), T_0), (mark(a, b), T_0), (mark(a, c), T_1), (mark(a, d), T_2), (mark(b, c), T_0), (mark(c, d), T_1), (mark(e, f), T_2), (reach(a, b), \mathbf{T}_0), (reach(a, c), \mathbf{T}_1), (reach(a, d), \mathbf{T}_2), (reach(b, c), \mathbf{T}_0), (reach(c, d), \mathbf{T}_1), (reach(e, f), \mathbf{T}_2).$
4. $(edge(a, b, \implies), T_0), (edge(a, c, \longrightarrow), T_0), (edge(a, d, \dashrightarrow), T_0), (edge(b, c, \implies), T_0), (edge(c, d, \longrightarrow), T_0), (edge(e, f, \dashrightarrow), T_0), (mark(a, b), T_0), (mark(a, c), T_1), (mark(a, d), T_2), (mark(b, c), T_0), (mark(c, d), T_1), (mark(e, f), T_2), (reach(a, b), T_0), (reach(a, c), \mathbf{T}_0), (reach(a, d), \mathbf{T}_1), (reach(b, c), T_0), (reach(c, d), T_1), (reach(e, f), T_2), (reach(b, d), \mathbf{T}_1).$
5. $(edge(a, b, \implies), T_0), (edge(a, c, \longrightarrow), T_0), (edge(a, d, \dashrightarrow), T_0), (edge(b, c, \implies), T_0), (edge(c, d, \longrightarrow), T_0), (edge(e, f, \dashrightarrow), T_0), (mark(a, b), T_0), (mark(a, c), T_1), (mark(a, d), T_2), (mark(b, c), T_0), (mark(c, d), T_1), (mark(e, f), T_2), (reach(a, b), T_0), (reach(a, c), T_0), (reach(a, d), T_1), (reach(b, c), T_0), (reach(c, d), T_1), (reach(e, f), T_2), (reach(b, d), T_1).$

Κεφάλαιο 6

Σχετικές προσεγγίσεις στη βιβλιογραφία

Στο κεφάλαιο αυτό θα εξετάσουμε τις εναλλακτικές προσεγγίσεις που υπάρχουν στη βιβλιογραφία σε σχέση με τη δική μας δουλειά. Στην πρώτη ενότητα θα διερευνήσουμε τη σχέση μεταξύ απειρότιμης λογικής και τη λογική του von Wright για έκφραση προτιμήσεων. Στη συνέχεια θα εξετάσουμε διάφορες προσεγγίσεις που έχουν προταθεί στο πεδίο του λογικού προγραμματισμού με προτιμήσεις. Τέλος, θα παρουσιάσουμε κάποια άρθρα σχετικά με την απειρότιμη λογική.

6.1 Λογική με προτιμήσεις

Η έννοια της προτίμησης παίζει ένα πολύ σημαντικό ρόλο σε μια πληθώρα πεδίων από την ηθική φιλοσοφία, την ψυχολογία, την κοινωνιολογία μέχρι τη θεωρία αποφάσεων και τα οικονομικά [23]. Παρόλο που η προτίμηση είχε μελετηθεί αποσπασματικά σε αυτά τα πεδία από τις αρχές του 20ού αιώνα, ο πρώτος που μελέτησε τυπικά τις προτιμήσεις μέσω της μαθηματικής λογικής ήταν ο Φινλανδός φιλόσοφος G. H. von Wright [48]. Η λογική που όρισε είναι μια προτασιακή λογική στην οποία ενσωματώνεται μια διμελής σχέση P τέτοια ώστε το pPq δηλώνει ότι η κατάσταση p προτιμάται από την κατάσταση q . Ο von Wright όρισε ένα σύστημα πέντε αξιωμάτων που περιγράφουν τις ιδιότητες που πρέπει να έχει η σχέση προτίμησης.

Το πέμπτο αξίωμα είναι το λεγόμενο “*ceteris paribus*”, που μεταφράζεται κυριολεκτικά ως «άλλα πράγματα το ίδιο πράγμα». Αυτό σημαίνει ότι μια τρίτη κατάσταση r δεν επιδρά καθόλου στη σχέση μεταξύ των δύο καταστάσεων p, q που μας ενδιαφέρουν. Για παράδειγμα, αν κάποιος προτιμάει ένα στρογγυλό τραπέζι από ένα τετράγωνο, θα προτιμάει ένα μαύρο στρογγυλό τραπέζι από ένα μαύρο τετράγωνο τραπέζι. Το

αξίωμα αυτό είναι το πιο ισχυρό από τα υπόλοιπα, και ήταν αυτό που δημιούργησε τα περισσότερα προβλήματα στη σημασιολογία της λογικής αυτής [37]. Μέχρι σήμερα έχουν προταθεί πολλοί εναλλακτικοί ορισμοί του “*ceteris paribus*”, οι περισσότεροι από τους οποίους είναι εκφρασμένοι σε τροπικές λογικές [18], [26].

Στη δική μας προσέγγιση, αντίθετα με αυτή του von Wright, θεωρούμε ότι ανάμεσα σε δύο γεγονότα υπάρχει πάντα σύγκριση ως προς την προτίμηση (ολικότητα). Επιστρέφοντας στο προηγούμενο παράδειγμα, θα υπάρχει πάντα σύγκριση μεταξύ οποιωνδήποτε δυο ιδιοτήτων του τραπέζιού. Αν αν κάποιος προτιμάει ένα στρογγυλό τραπέζι από ένα μαύρο τραπέζι, τότε (όπως και πριν) θα προτιμάει ένα μαύρο στρογγυλό τραπέζι από ένα μαύρο τετράγωνο τετράγωνο τραπέζι. Αν όμως προτιμάει ένα μαύρο τραπέζι από ένα στρογγυλό τραπέζι, τότε ένα μαύρο στρογγυλό τραπέζι θα είναι το ίδιο με ένα μαύρο τετράγωνο τετράγωνο τραπέζι. Φαίνεται ότι η ιδιότητα της ολικότητας που έχει το σύστημα προτιμήσεων της απειρότιμης λογικής είναι ασύμβατη με το “*ceteris paribus*”, τουλάχιστον έτσι όπως αυτό ερμηνεύεται στην αρχική προσέγγιση του von Wright.

6.2 Λογικός προγραμματισμός με προτιμήσεις

Η εισαγωγή προτιμήσεων στον λογικό προγραμματισμό έχει προταθεί από αρκετά άρθρα στη βιβλιογραφία. Οι προσεγγίσεις αυτές μπορούν να χωριστούν σε δύο κατηγορίες, ανάλογα με τον σκοπό που επιτελεί κάθε φορά η εισαγωγή των προτιμήσεων.

- Στην πρώτη κατηγορία, η εισαγωγή προτιμήσεων γίνεται με σκοπό τη μοντελοποίηση και την επίλυση προβλημάτων βελτιστοποίησης.
- Στη δεύτερη κατηγορία, η εισαγωγή προτιμήσεων γίνεται με σκοπό τον περιορισμό των προβλημάτων και των ασαφειών που προκύπτουν στις μη μονοτονικές επεκτάσεις του λογικού προγραμματισμού.

Στη συνέχεια θα εξετάσουμε κάποιες προσεγγίσεις που υπάρχουν σε καθεμιά από τις παραπάνω κατηγορίες και θα διερευνήσουμε τη σχέση που υπάρχει μεταξύ των προσεγγίσεων αυτών και της δικιάς μας προσέγγισης.

6.2.1 Προτιμήσεις για προβλήματα βελτιστοποίησης

Προβλήματα βελτιστοποίησης συναντώνται σε πολλές εφαρμογές. Σε ένα πρόβλημα βελτιστοποίησης αναζητείται η βέλτιστη λύση από ένα σύνολο λύσεων με βάση κάποιο κριτήριο βελτιστοποίησης, το οποίο συνήθως είναι μια αντικειμενική συνάρτηση. Δυ-

στυχώς όμως, πολλά προβλήματα βελτιστοποίησης έχουν σύνθετη συνάρτηση βελτιστοποίησης ή είναι δύσκολος ο ορισμός της. Συνεπώς, στα [19], [12], [20] προτείνεται μια επέκταση του λογικού προγραμματισμού στην οποία το κριτήριο βελτιστοποίησης ορίζεται με τη βοήθεια μιας σχέσης προτίμησης ανάμεσα σε κάποια κατηγορήματα βελτιστοποίησης. Ένα τέτοιο πρόγραμμα είναι το εξής:

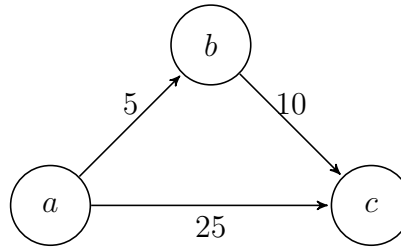
$$edge(a, b, 5). \quad edge(b, c, 10). \quad edge(a, c, 25).$$

$$path(X, Y, C) \leftarrow edge(X, Y, C).$$

$$path(X, Y, C_1 + C_2) \leftarrow path(X, Z, C_1), \quad edge(Z, Y, C_2).$$

$$path(X, Y, C_1) \succ path(X, Y, C_2) \leftarrow C_1 < C_2.$$

Διαισθητικά, στο πρόγραμμα αυτό το κατηγορήμα $path(X, Y, C)$ είναι αληθές αν το συντομότερο μονοπάτι μεταξύ των κόμβων X, Y είναι ίσο με C στον παρακάτω γράφο:



Αν αγνοήσουμε τον τελευταίο κανόνα του προγράμματος, που είναι ο κανόνας βελτιστοποίησης, το πρόγραμμα έχει ελάχιστο Herbrand μοντέλο το εξής:

$$M_C = \left\{ \begin{array}{l} edge(a, b, 5), \quad edge(b, c, 10), \quad edge(a, c, 25), \\ path(a, b, 5), \quad path(b, c, 10), \quad path(a, c, 25), \quad path(a, c, 15) \end{array} \right\}$$

ενώ το νόημα που θα δίνουμε στο πρόγραμμα αν συμπεριλαμβάναμε και τον κανόνα βελτιστοποίησης («προτιμητέο υπομοντέλο»), είναι το εξής υποσύνολο του M_C :

$$M_P = \left\{ \begin{array}{l} edge(a, b, 5), \quad edge(b, c, 10), \quad edge(a, c, 25), \\ path(a, b, 5), \quad path(b, c, 10), \quad path(a, c, 15) \end{array} \right\}$$

Ανάμεσα στα τρία άρθρα υπάρχουν αρκετές διαφορές, τόσο στο συντακτικό όσο και στη σημασιολογία. Στο [19] δίνεται μια σημασιολογία πιθανών κόσμων. Ένα κατηγορήμα βελτιστοποίησης (εδώ το $path$) μπορεί είτε να ανήκει είτε όχι σε κάποιο κόσμος, ενώ σαν νόημα του προγράμματος επιλέγεται ο προτιμότερος από αυτούς τους πιθανούς κόσμους. Επειδή η διαδικασία εύρεσης του πιθανότερου κόσμου δεν είναι

πρακτικά υπολογίσιμη, στο [12] προτείνεται ο μετασχηματισμός ενός προγράμματος με προτιμήσεις σε ένα ισοδύναμο λογικό πρόγραμμα με άρνηση, του οποίου το τρίτιμο well-founded μοντέλο είναι το νόημα του αρχικού προγράμματος. Στο [20] η σημασιολογία ορίζεται με χρήση δύο τελεστών ϕ, π οι οποίοι εφαρμόζονται στο ελάχιστο Herbrand μοντέλο M και διαγράφουν αρχικά τα λιγότερο προτιμητέα άτομα και στη συνέχεια κάθε άτομο που εξαρτάται από πρόσφατα διεγραμμένο άτομο. Το νόημα του προγράμματος δηλαδή είναι το $\pi^\alpha(\phi(M))$, που έχει σταθερό σημείο.

Ας συγκρίνουμε τις προσεγγίσεις αυτές με τη δικιά μας προσέγγιση. Η βασική ομοιότητα με τη δικιά μας προσέγγιση είναι ότι και στις δυο περιπτώσεις επεκτείνονται τα κλασικά λογικά προγράμματα (χωρίς άρνηση) με εισαγωγή προτιμήσεων μεταξύ ατόμων. Η βασική διαφορά είναι ότι στην προσέγγιση αυτή αφαιρούνται από το ελάχιστο Herbrand μοντέλο τα άτομα που είναι λιγότερο προτιμητέα σύμφωνα με τις κεφαλές των κανόνων βελτιστοποίησης. Αντίθετα, στη δικιά μας προσέγγιση, στο ελάχιστο απειρότιμο μοντέλο που υπολογίζεται τα λιγότερο προτιμητέα άτομα έχουν τιμές αληθείας που πλησιάζουν προς την κεντρική τιμή 0, ενώ η σχέση προτίμησης καθορίζεται στο σώμα των κανόνων. Για παράδειγμα, έστω τα παρακάτω προγράμματα που εκφράζουν την προτίμηση του $p(1)$ σε σχέση με το $p(2)$.

$$P_1 = \left\{ \begin{array}{l} p(1) \succ p(2). \\ p(2) \leftarrow p(1). \\ p(1). \end{array} \right\} \quad P_2 = \left\{ \begin{array}{l} p(2) \leftarrow \omega p(1). \\ p(1). \end{array} \right\}$$

Το προτιμητέο υπομοντέλο του P_1 είναι το $\{p(1)\}$, ενώ το ελάχιστο απειρότιμο μοντέλο του P_2 είναι το $\{(p(1), T_0), (p(2), T_1)\}$.

Δυνατότητα έκφρασης προτιμήσεων προσφέρεται και από διάφορες επεκτάσεις του λογικού προγραμματισμού με περιορισμούς [25]. Στην περιοχή αυτή συναντάμε κυρίως δύο προσεγγίσεις. Στην πρώτη προσέγγιση, σε κάθε κατηγορία που θέλουμε να βελτιστοποιήσουμε αντιστοιχίζουμε μια συνάρτηση, η βελτιστοποίηση της οποίας θα γίνει με την χρήση κάποιου τελεστή της μορφής *minimize* ή *maximize* [33]. Οι τελεστές αυτοί δεν ανήκουν στο «λογικό τμήμα» των συστημάτων που τους χρησιμοποιούν και η βελτιστοποίηση γίνεται συνήθως με την χρήση branch-and-bound μεθόδου. Στη δεύτερη προσέγγιση έχουμε τον λογικό προγραμματισμό με ιεραρχικούς περιορισμούς [51]. Εδώ ο προγραμματιστής μπορεί να δηλώνει ισχυρούς και αδύναμους περιορισμούς, έτσι ώστε από δύο λύσεις που ικανοποιούν τους ισχυρούς περιορισμούς να μπορεί να επιλεγεί από το σύστημα η λύση που ικανοποιεί τους αδύναμους περιορισμούς ως πιο προτιμότερη. Καθεμιά από τις δύο προσεγγίσεις έχει τα πλεονεκτήματά και τα μειονεκτήματά της. Η πρώτη μέθοδος είναι χρήσιμη όταν για το πρόβλημα μπορεί να

οριστεί μια συνάρτηση βελτιστοποίησης (κάτι που όμως δεν μπορεί να γίνει στη γενική περίπτωση) ενώ η δεύτερη μέθοδος προσφέρει καλύτερη εκφραστικότητα στις προτιμήσεις (αλλά ο υπολογισμός της καλύτερης λύσης μπορεί σε κάποιες περιπτώσεις να γίνει μη αποδοτικά).

6.2.2 Προτιμήσεις για μη μονοτονικότητα

Η εισαγωγή άρνησης στα σώματα των κανόνων οδηγεί σε μια μη μονοτονική επέκταση του λογικού προγραμματισμού, και πλέον δεν έχουμε ελάχιστο μοντέλο με τη σημασιολογία των van Emden και Kowalski. Οι δυο σημασιολογίες που έχουν επικρατήσει σε αυτήν την επεκτεταμένη κατηγορία λογικών προγραμμάτων είναι από τη μια τα well-founded semantics [47] και από την άλλη τα stable model semantics [16] και μια επέκταση των stable models, τα answer set semantics [17]. Η μεν πρώτη προσέγγιση επιστρέφει ένα ελάχιστο μοντέλο αλλά είναι τρίτιμο, δηλαδή περιέχει και την τιμή αληθείας 0 (που σημαίνει δεν γνωρίζω), ενώ η δεύτερη προσέγγιση επιστρέφει πολλά «ελαχιστικά» δίτιμα μοντέλα, αλλά κανένα από αυτά δεν είναι ελάχιστο.

Υπάρχει μια σειρά άρθρων που εισάγει προτιμήσεις στα λογικά προγράμματα ώστε να αντιμετωπισθούν τα προβλήματα αυτά, κυρίως επιλέγοντας κάποια από τα ελαχιστικά μοντέλα (answer sets) που είναι προτιμότερα. Για παράδειγμα, έστω το πρόγραμμα $P = \{a \leftarrow \text{not } b, b \leftarrow \text{not } a\}$. Το πρόγραμμα έχει δύο answer sets, τα $AS(P) = \{\{a\}, \{b\}\}$. Αν όμως θεωρήσουμε ότι το a είναι προτιμότερο από το b ή αν ο πρώτος κανόνας είναι προτιμότερος από τον δεύτερο τότε θα πάρουμε ως απάντηση το answer set $\{a\}$, και κατά συνέπεια να έχουμε ως απάντηση ένα δίτιμο μοντέλο αντί για δύο.

Μια πρώτη διάκριση στις προτιμήσεις είναι το αν είναι μεταξύ κανόνων ή μεταξύ ατόμων. Στο [43] εξετάζονται προτιμήσεις ανάμεσα στα άτομα, ενώ στα [50], [7] εξετάζονται προτιμήσεις ανάμεσα στους κανόνες. Και στις τρεις προσεγγίσεις αυτές εξετάζονται στατικές προτιμήσεις, δηλαδή η έκφραση των προτιμήσεων γίνεται μέσω μιας σχέσης διάταξης ανάμεσα στα άτομα ή τους κανόνες. Δυναμικές προτιμήσεις μεταξύ κανόνων εξετάζονται στα [6], [52], [13]. Οι προτιμήσεις εδώ μπορούν να βρίσκονται σε κεφαλές κανόνων, και με αυτόν τον τρόπο ορίζουμε ότι κάποιος κανόνας είναι προτιμότερος κάποιου άλλου όχι πάντα, αλλά όταν ισχύουν κάποιες προϋποθέσεις. Στο [15] προτείνεται ένας συνδυασμός δυναμικών προτιμήσεων μεταξύ κανόνων και ατόμων.

Όλες οι προσεγγίσεις ορίζονται πάνω στη σημασιολογία των answer sets, εκτός από το [6] που ορίζεται πάνω στα well-founded semantics. Επίσης όλες οι προσεγγίσεις αφορούν λογικά προγράμματα με άρνηση, εκτός από το [43] που αφορά διαζευκτικά λογικά προγράμματα με άρνηση και το [15] που αφορά απαγωγικά λογικά προγράμ-

ματα με άρνηση.

Καθεμιά από αυτές τις προσεγγίσεις ακολουθεί τον δικό της ορισμό για το τι είναι ένα προτιμότερο μοντέλο. Πώς όμως αυτές οι προσεγγίσεις συγκρίνονται μεταξύ τους; Στο [44] επιχειρείται μια σύγκριση μεταξύ των προσεγγίσεων των [50], [7] και [13] για εισαγωγή στατικών προτιμήσεων μεταξύ κανόνων πάνω στα answer set semantics (αν και το [13] μπορεί να υποστηρίξει και δυναμικές προτιμήσεις). Στο [44] η καθεμιά από τις τρεις προσεγγίσεις ορίζεται εκ νέου ώστε να υπάρχει συμβατότητα στους ορισμούς και αποδεικνύεται ότι τα προτιμότερα answer sets των προσεγγίσεων αυτών σχηματίζουν. Στο [14] υπάρχει μια σύγκριση πολύ περισσότερων προσεγγίσεων σχετικά με προτιμήσεις και μη-μονοτονικότητα, και όχι μόνο πάνω σε λογικά προγράμματα.

Οι προσεγγίσεις που υπάρχουν σε αυτήν την κατηγορία δεν συγκρίνονται άμεσα με τη δικιά μας προσέγγιση για διάφορους λόγους. Καταρχήν, οι προσεγγίσεις αυτές εστιάζουν κυρίως στα λογικά προγράμματα με άρνηση, ενώ εμείς επεκτείνουμε τα κλασικά λογικά προγράμματα (χωρίς άρνηση). Επίσης, οι προσεγγίσεις αυτές (αν εξαιρέσουμε το [43]) αφορούν προτιμήσεις μεταξύ κανόνων. Τέλος, ο λόγος που γίνεται η εισαγωγή των προτιμήσεων στις προσεγγίσεις αυτές είναι η αποφυγή προβλημάτων μη μονοτονικότητας, κάτι που δεν ισχύει στη δική μας προσέγγιση.

6.3 Απειρότιμη λογική

Όπως αναφέραμε και στο Κεφάλαιο 3, η απειρότιμη λογική ορίστηκε αρχικά στο [42], και προκύπτει από τον πολλαπλασιασμό των λογικών τιμών F, T με διαδοχικές δυνάμεις του απειροστού ϵ , έτσι ώστε να προκύψουν άπειρες τιμές αλήθειας. Κάθε τιμή αληθείας είναι «απείρως πιο σημαντική» από την επόμενη της, ενώ στο κέντρο βρίσκεται η τιμή $\epsilon^\infty \cdot T = \epsilon^\infty \cdot F = 0$, που σημαίνει «δεν γνωρίζω».

Η σύνδεση προτιμήσεων με την απειρότιμη λογική γίνεται στο [49] και στο [3], όπου εισάγονται επερωτήσεις με προτιμήσεις που χρησιμοποιούν τους τελεστές μ, ω . Οι επερωτήσεις αυτές είναι ίδιες με τα σώματα των κανόνων στη δικιά μας προσέγγιση. Επίσης στα [10], [11] γίνεται σύνδεση προτιμήσεων και απειροστών. Όμως, αντίθετα με τη δική μας προσέγγιση, ανάλογα με τις προτιμήσεις του χρήστη αντιστοιχίζεται σε κάθε αντικείμενο ένας αριθμός που περιέχει απειροστά (π.χ. $3 + 2\epsilon + 4\epsilon^2$), και όχι κάποια αληθοτιμή από το σύνολο άπειρων τιμών αληθείας.

Στο [42], η απειρότιμη λογική χρησιμοποιείται για τον ορισμό μοντελοθεωρητικής σημασιολογίας για τον λογικό προγραμματισμό με άρνηση μέσω αποτυχίας (συμβολίζεται με $\sim$). Έστω το πρόγραμμα $P = \{p, \quad q \leftarrow \sim r\}$. Το p είναι αληθές, το r είναι ψευδές και το q είναι αληθές. Διαισθητικά όμως, το q είναι «λιγότερο αληθές» από

το p , αφού η αλήθεια του p προκύπτει από τον πρώτο κανόνα, ενώ η αλήθεια του q προκύπτει επειδή δεν μπορέσαμε να αποδείξουμε το r . Συνεπώς, ο πίνακας αληθείας για τον τελεστή $\sim$ ορίζεται ως εξής:

$$\sim A = \begin{cases} T_{i+1}, & \text{αν } A = F_i \\ F_{i+1}, & \text{αν } A = T_i \\ 0, & \text{αν } A = 0 \end{cases}$$

Στην προσέγγιση αυτή, η διαβάθμιση στις τιμές αληθείας εκφράζει τον βαθμό «σιγουριάς» για κάποιο άτομο, δηλαδή από πόσες αρνήσεις εξαρτάται. Στη δικιά μας περίπτωση, η διαβάθμιση στις τιμές αληθείας εκφράζει τον βαθμό «προτίμησης» για κάποιο άτομο, δηλαδή την επίδραση που έχει η παρουσία ή η απουσία του.

Χρησιμοποιώντας την απειρότιμη λογική με παρόμοιο τρόπο όπως στο [42], στο [8] δίνεται μοντελοθεωρητική σημασιολογία για τον διαζευκτικό λογικό προγραμματισμό με άρνηση, ενώ στο [30] δίνεται σημασιολογία στις boolean γραμματικές, οι οποίες ορίζονται στο [40] και είναι μια επέκταση των γραμματικών χωρίς συμφραζόμενα που περιέχουν επιπλέον σύζευξη και άρνηση στα σώματα των κανόνων.

Εκτός από την απειρότιμη λογική υπάρχουν και άλλες μη κλασικές λογικές στις οποίες υπάρχουν άπειρες τιμές αληθείας, η σημαντικότερη από τις οποίες είναι η ασαφής λογική [21]. Στην ασαφή λογική η τιμή αληθείας είναι μια πραγματική τιμή στο διάστημα $[0, 1]$ η οποία εκφράζει τον βαθμό αλήθειας μιας πρότασης. Οι δύο λογικές διαφέρουν σημαντικά ως προς τη μορφή του συνόλου των αληθοτιμών, αφού στην απειρότιμη λογική έχουμε ένα άπειρο διακριτό σύνολο αληθοτιμών και όχι συνεχές διάστημα.

Κεφάλαιο 7

Παρατηρήσεις και μελλοντικές κατευθύνσεις

Στην παρούσα διπλωματική εργασία ορίσαμε μια νέα επέκταση του κλασικού λογικού προγραμματισμού. Αρχικά, εισάγαμε στα σώματα των κανόνων δύο τελεστές προτίμησης με τη σημασία «προαιρετικά» και «εναλλακτικά» και στη συνέχεια δώσαμε σημασιολογία σταθερού σημείου για τη νέα αυτή γλώσσα. Η σημασιολογία βασίζεται πάνω στην απειρότιμη λογική, μια επέκταση της κλασικής δίτιμης λογικής που χρησιμοποιεί άπειρες τιμές αληθείας οι οποίες βρίσκονται ενδιάμεσα στις δύο ακραίες *False*, *True*.

Στο τελευταίο αυτό κεφάλαιο θα συνοψίσουμε παραθέτοντας κάποιες παρατηρήσεις, καθώς και πιθανές μελλοντικές κατευθύνσεις για την έρευνα πάνω στη νέα αυτή γλώσσα.

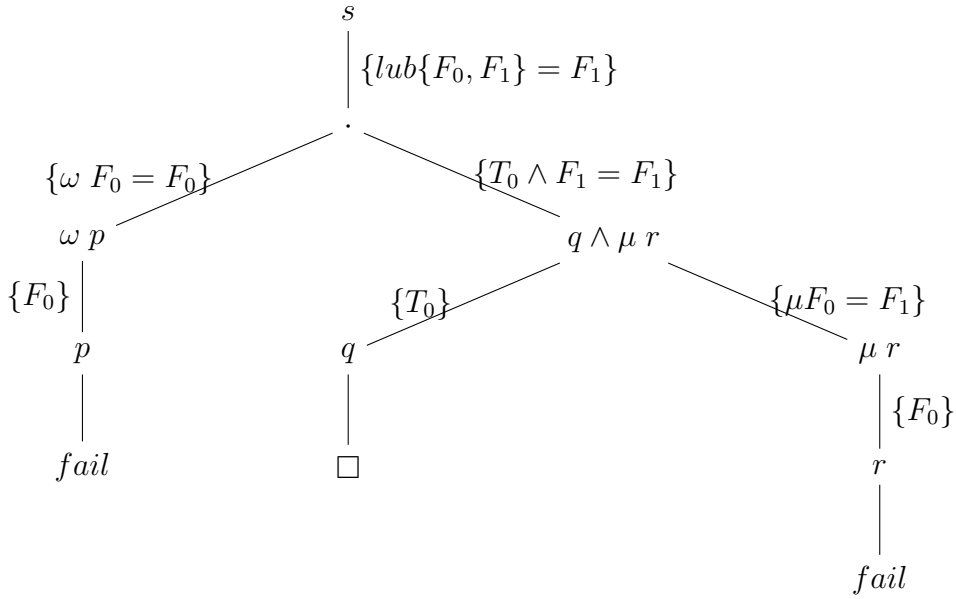
7.1 Λειτουργική σημασιολογία

Πέρα από τη δηλωτική σημασιολογία του λογικού προγραμματισμού με προτιμήσεις που δίνεται σε αυτήν την εργασία, θα μπορούσαμε στο μέλλον να δώσουμε και μια λειτουργική σημασιολογία, μέσω μιας διαδικασίας απόδειξης. Η διαδικασία απόδειξης θα είναι λογικά μια επέκταση της SLD-επίλυσης του κλασικού λογικού προγραμματισμού [32].

Ας δούμε ένα παράδειγμα για να εξετάσουμε πως περίπου μπορεί να μοιάζει μια τέτοια διαδικασία απόδειξης. Έστω το πρόγραμμα:

$$P = \left\{ s \leftarrow \omega p, \quad s \leftarrow q \wedge \mu r, \quad q \right\}$$

Η διαδικασία επίλυσης του στόχου $\leftarrow s$ μοιάζει ως εξής:



Συνεπώς, μια μελλοντική κατεύθυνση έρευνας για τον λογικό προγραμματισμό με προτιμήσεις είναι το να δοθεί λειτουργική σημασιολογία. Αυτό μπορεί να γίνει μέσω του ακριβούς ορισμού μιας διαδικασίας απόδειξης που μοιάζει με την SLD-επίλυση του κλασικού λογικού προγραμματισμού.

7.2 Στρωματοποίηση λογικών προγραμμάτων με προτιμήσεις

Η έννοια της στρωματοποίησης ορίζεται για τα λογικά προγράμματα με άρνηση. Ένα λογικό πρόγραμμα με άρνηση P , ονομάζεται:

- *στρωματοποιήσιμο* [4], αν δεν υπάρχει κάποιο κατηγορημα που να εξαρτάται κυκλικά από τον εαυτό του μέσω άρνησης.
- *τοπικά στρωματοποιήσιμο* [41], αν στο $ground(P)$ δεν υπάρχει κάποιο άτομο που να εξαρτάται κυκλικά από τον εαυτό του μέσω άρνησης.

Κάθε στρωματοποιήσιμο πρόγραμμα είναι και τοπικά στρωματοποιήσιμο. Ένα πρόγραμμα που είναι τοπικά στρωματοποιήσιμο και δεν είναι στρωματοποιήσιμο είναι το $P = \{even(0), even(s(X)) \leftarrow not\ even(X)\}$. Το κατηγορημα $even$ εξαρτάται από τον εαυτό του μέσω άρνησης, αλλά στο $ground(P)$ το άτομο $even(s^i(0))$ εξαρτάται μέσω άρνησης από το άτομο $even(s^{i-1}(0))$ και όχι από τον εαυτό του.

Με τον ίδιο τρόπο, μπορούμε να ορίσουμε την έννοια της στρωματοποίησης για τα λογικά προγράμματα με προτιμήσεις. Ένα λογικό πρόγραμμα με προτιμήσεις P , ονομάζεται:

- *στρωματοποιήσιμο*, αν δεν υπάρχει κάποιο κατηγορήμα που να εξαρτάται κυκλικά από τον εαυτό του μέσω των τελεστών προτίμησης μ, ω .
- *τοπικά στρωματοποιήσιμο*, αν στο $\text{ground}(P)$ δεν υπάρχει κάποιο άτομο που να εξαρτάται κυκλικά από τον εαυτό του μέσω των τελεστών προτίμησης μ, ω .

Όλα τα λογικά προγράμματα με προτιμήσεις που έχουμε δει μέχρι στιγμής είναι στρωματοποιήσιμα. Φαίνεται ότι για τα στρωματοποιήσιμα προγράμματα το ελάχιστο μοντέλο είναι πάντα δίτιμο. Διαισθητικά, έχουμε πεπερασμένο αριθμό κατηγορημάτων, κανένα κατηγορήμα δεν εξαρτάται κυκλικά από τον εαυτό του μέσω των τελεστών προτίμησης μ, ω , και οι τέλεστες μ, ω είναι ο μόνος τρόπος να πλησιάσουμε προς την κεντρική τιμή 0 ώστε το μοντέλο μας να γίνει τρίτιμο.

Στη συνέχεια, ας δούμε δύο περιπτώσεις τοπικά στρωματοποιήσιμων προγραμμάτων που δεν είναι στρωματοποιήσιμα. Τα προγράμματα

$$P_1 = \{p(s(X)) \leftarrow \mu p(X)\}, \quad P_2 = \{p(0), p(s(X)) \leftarrow \omega p(X)\}$$

έχουν ελάχιστα μοντέλα αντίστοιχα τα

$$M_{P_1} = \{(p(0), F_0), (p(s(0)), F_1), (p(s(s(0))), F_2), \dots (p(s^i(0)), F_i), \dots\}$$

$$M_{P_2} = \{(p(0), T_0), (p(s(0)), T_1), (p(s(s(0))), T_2), \dots (p(s^i(0)), T_i), \dots\}$$

Φαίνεται ότι και τα τοπικά στρωματοποιήσιμα προγράμματα με προτιμήσεις έχουν νόημα. Στη συνέχεια ας δούμε κάποια μη τοπικά στρωματοποιήσιμα προγράμματα.

$$P_1 = \{p \leftarrow \mu p\}, \quad P_2 = \{p, p \leftarrow \omega p\}$$

Για το P_1 έχουμε $T_{P_1} \uparrow 0 = \{(p, F_0)\}$, $T_{P_1} \uparrow 1 = \{(p, F_1)\}$, $T_{P_1} \uparrow 2 = \{(p, F_2)\}$, $T_{P_1} \uparrow 3 = \{(p, F_3)\}, \dots$ $T_{P_1} \uparrow \omega = \{(p, 0)\}$ άρα $M_{P_1} = \{(p, 0)\}$, ενώ για το P_2 έχουμε $T_{P_2} \uparrow 0 = \{(p, T_0)\}$, $T_{P_2} \uparrow 1 = \{(p, T_0)\}$ άρα $M_{P_2} = \{(p, T_0)\}$.

Παρατηρούμε μια ασύμμετρη συμπεριφορά στα δύο παραδείγματα. Στο πρώτο πρόγραμμα οι αληθοτιμές για το p αυξάνονται συνεχώς με την εφαρμογή του τελεστή μ μέχρι να φτάσουν στο 0, γιατί ισχύει ότι

$$I(p) = F_i \implies T_{P_1}(I)(p) = \text{lub}\{I(\mu p)\} = \text{lub}\{\mu I(p)\} = \text{lub}\{F_{i+1}\} = F_{i+1}$$

Αντίθετα, στο δεύτερο παράδειγμα, από τη στιγμή που για το p θα δοθεί η αληθοτιμή T_0 , δεν μπορεί να μειωθεί άλλο με την εφαρμογή του τελεστή ω γιατί ισχύει ότι

$$I(p) = T_0 \implies T_{P_2}(I)(p) = \text{lub}\{I(p), I(\omega p)\} = \text{lub}\{I(p), \omega I(p)\} = \text{lub}\{T_0, T_1\} = T_0$$

Όμως, δεν εμφανίζεται πάντα η αληθοτιμή 0 όταν υπάρχει κύκλος μέσω του μ στα μη τοπικά στρωματοποιήσιμα προγράμματα. Στο πρόγραμμα

$$P_1 = \{q(Y) \leftarrow p(X, Y) \wedge \mu q(X), \quad p(1, 2), \quad p(2, 3)\}$$

το ελάχιστο μοντέλο είναι το

$$M_{P_1} = \{(q(1), F_0), (q(2), F_1), (q(3), F_2), (q(4), F_3)\}$$

ενώ αν προσθέσουμε το γεγονός $p(3, 1)$, τότε στο πρόγραμμα

$$P_2 = \{q(Y) \leftarrow p(X, Y) \wedge \mu q(X), \quad p(1, 2), \quad p(2, 3), \quad p(3, 1)\}$$

το ελάχιστο μοντέλο είναι το

$$M_{P_2} = \{(q(1), 0), (q(2), 0), (q(3), 0), (q(4), 0)\}$$

Συνεπώς, μια μελλοντική κατεύθυνση έρευνας για τον λογικό προγραμματισμό με προτιμήσεις είναι το να εξεταστούν πιο τυπικά οι διάφορες κατηγορίες στρωματοποίησης και οι μορφές των μοντέλων που προκύπτουν σε κάθε περίπτωση.

7.3 Επεκτάσεις της γλώσσας

Η γλώσσα που προτείνουμε μπορεί να επεκταθεί με πολλούς τρόπους έτσι ώστε να γίνει πιο εκφραστική και με περισσότερες εφαρμογές. Μια πρώτη επέκταση θα μπορούσε να είναι η προσθήκη επιπλέον τελεστών. Μερικοί από αυτούς θα μπορούσαν να είναι:

- τελεστές αύξησης προτίμησης μ^{-1}, ω^{-1} , οι οποίοι θα μπορούσαν να είναι οι

$$\mu^{-1}A = \begin{cases} F_0, & \text{αν } A = F_0 \\ F_{i-1}, & \text{αν } A = F_i, i \geq 1 \\ T_i, & \text{αν } A = T_i \end{cases} \quad \omega^{-1}A = \begin{cases} T_0, & \text{αν } A = T_0 \\ T_{i-1}, & \text{αν } A = T_i, i \geq 1 \\ F_i, & \text{αν } A = F_i \end{cases}$$

- άρνηση (μέσω αποτυχίας) στο σώμα των κανόνων (τελεστής $\sim$), συνδυάζοντας κάπως αυτή τη δουλειά με το [42].
- κάποιους τελεστές σύγκρισης αληθοτιμών, σαν τους παρακάτω:

$$A \approx B = \begin{cases} T_0, & \text{αν } A = B \\ F_0, & \text{διαφορετικά} \end{cases} \quad A \not\approx B = \begin{cases} T_0, & \text{αν } A \neq B \\ F_0, & \text{διαφορετικά} \end{cases}$$

$$A \prec B = \begin{cases} T_0, & \text{αν } A < 0 < B \\ T_1, & \text{αν } A < B < 0 \text{ ή } 0 < A < B \\ 0, & \text{αν } A = B \\ F_1, & \text{αν } B < A < 0 \text{ ή } 0 < B < A \\ F_0, & \text{αν } B < 0 < A \end{cases}$$

Ο τελεστής $\prec$ είναι ένας ποσοτικός τελεστής σύγκρισης, αφού η τάξη της αληθοτιμής του αποτελέσματος μπορεί να εκφράσει αν οι A, B βρίσκονται εκατέρωθεν του 0 ή όχι.

- κάποιοι τελεστές που θα λειτουργούν σαν «φίλτρα» στις αληθοτιμές:

$$A \odot C_H = \begin{cases} A, & \text{αν } A \leq C_H \\ C_H, & \text{αν } A > C_H \end{cases} \quad A \odot C_L = \begin{cases} A, & \text{αν } A \geq C_L \\ C_L, & \text{αν } A < C_L \end{cases}$$

$$A \odot x = \begin{cases} A, & \text{αν } A \leq F_x \\ 0, & \text{αν } F_x < A < T_x \\ A, & \text{αν } A \geq T_x \end{cases}$$

Οι τελεστές αυτοί θα μπορούσαν να χρησιμοποιηθούν για να αγνοήσουμε την τιμή των αληθοτιμών από ένα σημείο και μετά.

Μια επιπλέον επέκταση της γλώσσας, θα μπορούσε να είναι επαύξηση του πεδίου των αληθοτιμών, ώστε να μπορούν να εκφραστούν 2 η περισσότεροι βαθμοί προτίμησης ταυτόχρονα. Για παράδειγμα, το νέο επαυξημένο σύνολο αληθοτιμών θα μπορούσε να έχει αυτή τη μορφή:

$$F_0 < F_{0+\epsilon} < F_{0+2\epsilon} < \dots < F_1 < F_{1+\epsilon} < F_{1+2\epsilon} < \dots < F_2 < F_{2+\epsilon} < F_{2+2\epsilon} < \dots < 0$$

$$0 < \dots < T_{2+2\epsilon} < T_{2+\epsilon} < T_2 < \dots < T_{1+2\epsilon} < T_{1+\epsilon} < T_1 < \dots < T_{0+2\epsilon} < T_{0+\epsilon} < T_0$$

Για παράδειγμα, κάποιος έχει τις εξής προτιμήσεις: προτιμάει καφέ από τσάι, και προτιμάει μπύρα από κρασί αλλά όταν η έξοδος είναι βραδυνή προτιμάει αλκοολούχο ποτό από μη αλκοολούχο. Σε αυτήν την περίπτωση λοιπόν θα μπορούσαμε να αναθέσουμε στα ποτά μπύρα, κρασί, καφέ και τσάι τις τιμές $T_0, T_{0+\epsilon}, T_1, T_{1+\epsilon}$ αντίστοιχα, ώστε να εκφραστεί καλύτερα η προτίμηση σε αλκοολούχα ποτά.

Επίσης, η επάυξηση του πεδίου αληθοτιμών θα μπορούσε ενδεχομένως να βοηθήσει στην εισαγωγή άρνησης στον λογικό προγραμματισμό με προτιμήσεις. Η πρώτη διαβάθμιση στις τιμές αληθείας θα μπορούσε να σχετίζεται με τον βαθμό «σιγουριάς» ενός ατόμου (όπως στο [42]), ενώ η δεύτερη διαβάθμιση με τον βαθμό «προτίμησης» ενός ατόμου.

7.4 Προς μια απειρότιμη θεωρία συνόλων

Στην κλασική θεωρία συνόλων ένα στοιχείο e είτε ανήκει είτε δεν ανήκει σε ένα σύνολο S . Χρησιμοποιώντας συμβολισμούς αντίστοιχους με τον συναρτησιακό προγραμματισμό, μπορούμε να πούμε ότι για κάθε e είναι είτε $S(e) = True$ είτε $S(e) = False$. Επεκτείνοντας αυτήν την ιδέα, μπορούμε να πούμε ότι κάθε στοιχείο e ανήκει ή δεν ανήκει μερικώς σε ένα σύνολο S , δηλαδή $S(e) = v$, με $v \in \{F_0, T_0, F_1, T_1, \dots\}$.

Κάνοντας ένα επόμενο βήμα, θα πρέπει να ορίσουμε τις βασικές πράξεις μεταξύ των απειρότιμων συνόλων, οι οποίες είναι η σχέση υποσυνόλου, τομής και ένωσης. Η σχέση υποσυνόλου είναι μια σχέση μερικής διάταξης, οπότε μπορούμε να θεωρήσουμε τη σχέση $\leq$ (ορισμός 4.11) ως το ευθύ ανάλογο της σχέσης υποσυνόλου στα κλασικά σύνολα. Επειδή για κάθε οικογένεια συνόλων $\mathcal{S}$ ορίζονται τα $\text{lub}(\mathcal{S})$ και $\text{glb}(\mathcal{S})$ (λήμμα 4.2) και επειδή για κάθε $S \in \mathcal{S}$ είναι $\text{lub}(\mathcal{S}) \geq S$ και $\text{glb}(\mathcal{S}) \leq S$ μπορούμε να συμπεράνουμε ότι το $\text{lub}\{S_1, S_2\}$ είναι το ευθύ ανάλογο της ένωσης $S_1 \cup S_2$ ενώ το $\text{glb}\{S_1, S_2\}$ είναι το ευθύ ανάλογο της τομής $S_1 \cap S_2$.

Το παρακάτω παράδειγμα δείχνει ότι τα παραπάνω γενικεύουν αυτά που ξέρουμε για τα κλασικά σύνολα:

Παράδειγμα 7.1. Έστω $A = \{(a, T_0), (b, T_0), (c, F_0)\}$, $B = \{(a, F_0), (b, T_0), (c, T_0)\}$, $E = \{(a, F_0), (b, F_0), (c, F_0)\}$. Είναι $E \leq A$ και $E \leq B$ και $\text{glb}\{A, B\} = \{(a, F_0), (b, T_0), (c, F_0)\}$ και $\text{lub}\{A, B\} = \{(a, T_0), (b, T_0), (c, T_0)\}$.

Η σχέση $\leq$, παρόλο που μας βοήθησε στον ορισμό της σημασιολογίας των λογικών προγραμμάτων με προτιμήσεις, έχει ένα σημαντικό μειονέκτημα. Δεν εκμεταλλεύεται το ότι οι τιμές επιπέδου 0 είναι απείρως σημαντικότερες από τις τιμές επιπέδου 1 κ.ο.κ., ώστε να επιτυγχάνεται καλύτερη σύγκριση. Στο [42] ορίζεται η σχέση μερικής

διάταξης $\sqsubseteq_\infty$ που κάνει διαισθητικά ακριβώς αυτό: συγκρίνει τα στοιχεία με αληθοτιμές ανα επίπεδο και δεν ξετάζει τις τιμές πέρα από το σημείο στο οποίο παρατηρηθεί η διαφορά. Αυτό φαίνεται στο παρακάτω παράδειγμα:

Παράδειγμα 7.2. Έστω $A = \{(a, T_0), (b, F_1), (c, T_1), (d, T_2), (e, F_2)\}$ και $B = \{(a, T_0), (b, T_1), (c, T_1), (d, F_2), (e, F_2)\}$. Είναι $A \not\sqsubseteq B$ και $B \not\sqsubseteq A$, γιατί $A(c) \leq B(c)$ και $A(d) \geq B(d)$, ενώ $A \sqsubseteq_\infty B$, γιατί η σύγκριση των αληθοτιμών θα γίνει μέχρι το επίπεδο 1.

Παρατηρούμε ότι η σχέση $\leq$ είναι ειδική περίπτωση της σχέσης $\sqsubseteq_\infty$. Πιο τυπικά αυτό αποδεικνύεται στο [42] (λήμμα 4.13).

Το ερώτημα για μελλοντική έρευνα στα απειρότιμα σύνολα είναι το εξής: ενώ αν θεωρήσουμε ως σχέση υποσυνόλου τη σχέση $\leq$ προκύπτουν εύκολα τα ανάλογα της τομής και της ένωσης (glb και lub αντίστοιχα), αν θεωρήσουμε ως σχέση υποσυνόλου τη σχέση $\sqsubseteq_\infty$ ποιά είναι τα ανάλογα της τομής και της ένωσης;

	απλά σύνολα	απειρότιμα σύνολα	
υποσύνολο	$\subseteq$	$\leq$	$\sqsubseteq_\infty$
τομή	$\cap$	glb	?
ένωση	$\cup$	lub	?

Γλωσσάρι

Ελληνο-αγγλικό

- απαγωγικός λογικός προγραμματισμός - abductive logic programming
- απειροστό - infinitesimal
- απειρότιμη ερμηνεία - infinite-valued interpretation
- απειρότιμη λογική - infinitesimal logic
- απειρότιμο μοντέλο - infinite-valued model
- άτομο - atom
- βάση Herbrand - Herbrand base
- διαδικασία απόδειξης - proof procedure
- διαζευκτικός λογικός προγραμματισμός - disjunctive logic programming
- διατακτικός αριθμός - ordinal number
- επαγωγικός λογικός προγραμματισμός - inductive logic programming
- επόμενος διατακτικός - successor ordinal
- ερμηνεία Herbrand - Herbrand interpretation
- θεωρία συνόλων - set theory
- κανόνας - clause
- λειτουργική σημασιολογία - operational semantics
- λογική με προτιμήσεις - preference logic
- λογικό πρόγραμμα - logic program
- λογικός προγραμματισμός - logic programming
- λογικός προγραμματισμός με ιεραρχικούς περιορισμούς - hierarchical constraint logic programming
- λογικός προγραμματισμός με περιορισμούς - constraint logic programming
- λογικός προγραμματισμός υψηλής τάξης - higher-order logic programming
- μονοτονικός τελεστής - monotonic operator
- μοντέλο Herbrand - Herbrand model
- μοντελοθεωρητική σημασιολογία - model-theoretic semantics
- οριακός διατακτικός - limit ordinal
- όρος - term

πιθανοτικός λογικός προγραμματισμός - probabilistic logic programming
πλήρες πλέγμα - complete lattice
πλήρης αποτίμηση - ground instantiation
πρόβλημα βελτιστοποίησης - optimization problem
προτιμήσεις - preferences
σημασιολογία - semantics
σημασιολογία σταθερού σημείου - fixpoint semantics
σταθερό σημείο - fixed point
στόχος - goal
στρωματοποιήσιμο - stratified
σύμπαν Herbrand - Herbrand universe
συναρτησιακός προγραμματισμός - functional programming
συνεχής τελεστής - continuous operator
συντακτικό - syntax
σχέση ανακλαστική - reflexive relation
σχέση αντισυμμετρική - antisymmetric relation
σχέση μερικής διάταξης - partial order relation
σχέση μεταβατική - transitive relation
σχέση ολική - total relation
σχέση ολικής διάταξης - total order relation
τάξη - order
τελεστής - operator
τελεστής άμεσου επακόλουθου - immediate consequence operator
τοπικά στρωματοποιήσιμο - locally stratified
φράγμα - bound
χρονικός λογικός προγραμματισμός - temporal logic programming
SLD-επίλυση - SLD-resolution

Αγγλο-ελληνικό

abductive logic programming - απαγωγικός λογικός προγραμματισμός
antisymmetric relation - σχέση αντισυμμετρική
atom - άτομο
bound - φράγμα
clause - κανόνας
complete lattice - πλήρες πλέγμα
constraint logic programming - λογικός προγραμματισμός με περιορισμούς
continuous operator - συνεχής τελεστής

disjunctive logic programming - διαζευκτικός λογικός προγραμματισμός
fixed point - σταθερό σημείο
fixpoint semantics - σημασιολογία σταθερού σημείου
functional programming - συναρτησιακός προγραμματισμός
goal - στόχος
ground instantiation - πλήρης αποτίμηση
Herbrand base - βάση Herbrand
Herbrand interpretation - ερμηνεία Herbrand
Herbrand model - μοντέλο Herbrand
Herbrand universe - σύμπαν Herbrand
hierarchical constraint logic programming - λογικός προγραμματισμός με ιεραρχικούς περιορισμούς
higher order logic programming - λογικός προγραμματισμός υψηλής τάξης
immediate consequence operator - τελεστής άμεσου επακόλουθου
inductive logic programming - επαγωγικός λογικός προγραμματισμός
infinitesimal logic - απειρότιμη λογική
infinitesimal - απειροστό
infinite-valued interpretation - απειρότιμη ερμηνεία
infinite-valued model - απειρότιμο μοντέλο
limit ordinal - οριακός διατακτικός
locally stratified - τοπικά στρωματοποιήσιμο
logic programming - λογικός προγραμματισμός
logic program - λογικό πρόγραμμα
modeltheoretic semantics - μοντελοθεωρητική σημασιολογία
monotonic operator - μονοτονικός τελεστής
operational semantics - λειτουργική σημασιολογία
operator - τελεστής
optimization problem - πρόβλημα βελτιστοποίησης
order - τάξη
ordinal number - διατακτικός αριθμός
partial order relation - σχέση μερικής διάταξης
preference logic - λογική με προτιμήσεις
preferences - προτιμήσεις
probabilistic logic programming - πιθανοτικός λογικός προγραμματισμός
proof procedure - διαδικασία απόδειξης
reflexive relation - σχέση ανακλαστική
semantics - σημασιολογία
set theory - θεωρία συνόλων
SLD-resolution - SLD-επίλυση
stratified - στρωματοποιήσιμο

successor ordinal - επόμενος διατακτικός

syntax - συντακτικό

temporal logic programming - χρονικός λογικός προγραμματισμός

term - όρος

total order relation - σχέση ολικής διάταξης

total relation - σχέση ολική

transitive relation - σχέση μεταβατική

Βιβλιογραφία

- [1] M. Abadi and Z. Manna. Temporal logic programming. In *International Symposium on Logic Programming*, pages 4–16, San Francisco, CA, Sept. 1987. IEEE.
- [2] S. Abramsky and A. Jung. Domain theory. In S. Abramsky, D. Gabbay, and T. S. E. Maibaum, editors, *Handbook of Logic in Computer Science Volume 3*, pages 1–168. Oxford University Press, 1994.
- [3] R. Agarwal. A framework for expressing prioritized constraints using infinitesimal logic. Master’s thesis, University of Victoria, 2005.
- [4] K. Apt, H. A. Blair, and A. Walker. Towards a theory of declarative knowledge. In J. Minker, editor, *Foundations of deductive databases and logic programming*, pages 89–142, Los Altos, CA, 1988. Morgan Kaufmann.
- [5] K. R. Apt. *From logic programming to Prolog*. Prentice-Hall, Inc., Upper Saddle River, NJ, 1996.
- [6] G. Brewka. Well-founded semantics for extended logic programs with dynamic preferences. *J. Artif. Intell. Res. (JAIR)*, 4:19–36, 1996.
- [7] G. Brewka and T. Eiter. Preferred answer sets for extended logic programs. In *KR*, pages 86–97, 1998.
- [8] P. Cabalar, D. Pearce, P. Rondogiannis, and W. W. Wadge. A purely model-theoretic semantics for disjunctive logic programs with negation. In C. Baral, G. Brewka, and J. S. Schlipf, editors, *Logic Programming and Nonmonotonic Reasoning, 9th International Conference, LPNMR 2007, Tempe, AZ, USA, May 15-17, 2007, Proceedings*, volume 4483 of *Lecture Notes in Computer Science*, pages 44–57. Springer, 2007.
- [9] A. Charalambidis, K. Handjopoulos, P. Rondogiannis, and W. W. Wadge. Extensional higher-order logic programming. *CoRR*, abs/1106.3457, 2011.
- [10] M. Chowdhury, A. Thomo, and W. W. Wadge. Preferential infinitesimals for information retrieval. In L. S. Iliadis, I. Maglogiannis, G. Tsoumakas, I. P. Vlahavas, and M. Bramer, editors, *AIAI*, volume 296 of *IFIP*, pages 113–125. Springer, 2009.

- [11] M. Chowdhury, A. Thomo, and W. W. Wadge. Trust-based infinitesimals for enhanced collaborative filtering. In S. Chawla, K. Karlapalem, and V. Pudi, editors, *COMOD*. Computer Society of India, 2009.
- [12] B. Cui and T. Swift. Preference logic grammars: Fixed point semantics and application to data standardization. *Artificial Intelligence*, 138(1–2):117–147, 2002.
- [13] J. P. Delgrande, T. Schaub, and H. Tompits. Logic programs with compiled preferences. In W. Horn, editor, *ECAI 2000, Proceedings of the 14th European Conference on Artificial Intelligence, Berlin, Germany, August 20-25, 2000*, pages 464–468. IOS Press, 2000.
- [14] J. P. Delgrande, T. Schaub, H. Tompits, and K. Wang. A classification and survey of preference handling approaches in nonmonotonic reasoning. *Computational Intelligence*, 20(2):308–334, 2004.
- [15] P. Dell’Acqua and L. M. Pereira. Preferential theory revision. *J. Applied Logic*, 5(4):586–601, 2007.
- [16] M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In *ICLP/SLP*, pages 1070–1080, 1988.
- [17] M. Gelfond and V. Lifschitz. Classical negation in logic programs and disjunctive databases. *New Generation Computing*, 9:365–385, 1991.
- [18] P. Girard. Von wright’s preference logic reconsidered, 2006.
- [19] K. Govindarajan, B. Jayaraman, and S. Mantha. Preference logic programming. In *ICLP*, pages 731–745, 1995.
- [20] H.-F. Guo and B. Jayaraman. Logic programming with solution preferences. *J. Log. Algebr. Program*, 78(1):1–21, 2008.
- [21] P. Hajek. Fuzzy logic. In E. N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Fall 2010 edition, 2010.
- [22] P. R. Halmos. *Naive Set Theory*. Springer, New York, 1974.
- [23] S. O. Hansson and T. GrÅEne-Yanoff. Preferences. In E. N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Fall 2011 edition, 2011.
- [24] J. Jaffar and J.-L. Lassez. Constraint logic programming. In *Symposium on Principles of Programming Languages*. Munich, 1987.
- [25] J. Jaffar and M. Maher. Constraint logic programming: A survey. *Journal of Logic Programming*, 19-20:503–581, 1994.

- [26] A. L. B. Jr., S. Mantha, and T. Wakayama. Preference logics: Towards a unified approach to nonmonotonicity in deductive reasoning. *Ann. Math. Artif. Intell*, 10(3):233–280, 1993.
- [27] A. C. Kakas, R. A. Kowalski, and F. Toni. Abductive Logic Programming. *Journal of Logic and Computation*, 2(6):719–770, 1993.
- [28] H. J. Keisler. *Foundations of Infinitesimal Calculus*. 2007.
- [29] S. C. Kleene. *Introduction to Metamathematics*. Van Nostrand, Princeton, NJ, 1952.
- [30] V. Kountouriotis, C. Nomikos, and P. Rondogiannis. Well-founded semantics for boolean grammars. *Inf. Comput*, 207(9):945–967, 2009.
- [31] R. Kowalski. Algorithms = logic + control. *Communications of the ACM (CACM)*, 22(7):424–436, 1979.
- [32] J. W. Lloyd. *Foundations of Logic Programming, Second Extended Edition*. Springer Verlag, 1993.
- [33] M. J. Maher and P. J. Stuckey. Expanding query power in constraint logic programming languages. In *NACLP*, pages 20–36, 1989.
- [34] D. Miller. Higher-order logic programming. In *ICLP*, page 784, 1990.
- [35] J. Minker and A. Rajasekar. A fixpoint semantics for disjunctive logic programs. *Journal of Logic Programming*, 9(1):45–74, July 1990.
- [36] S. Muggleton, editor. *Inductive Logic Programming*. Academic Press, 1992.
- [37] J. D. Mullen. Does the logic of preference rest on a mistake? *Metaphilosophy*, 10:247–255, 1979.
- [38] R. T. Ng and V. S. Subrahmanian. Probabilistic logic programming. *Information and Computation*, 101(2):150–201, 1992.
- [39] U. Nilsson and J. Maluszynski. *Logic, programming and Prolog*. Wiley, 1990.
- [40] A. Okhotin. Boolean grammars. *Inf. Comput*, 194(1):19–48, 2004.
- [41] T. Przymusiński. On the declarative semantics of deductive databases and logic programs. In *Foundations of deductive databases and logic programming*, pages 193–216, Los Altos, CA., 1988. Morgan Kaufmann.
- [42] P. Rondogiannis and W. W. Wadge. Minimum model semantics for logic programs with negation-as-failure. *ACM Trans. Comput. Log*, 6(2):441–467, 2005.
- [43] C. Sakama and K. Inoue. Representing priorities in logic programs. In *JICSLP*, pages 82–96, 1996.

- [44] T. Schaub and K. Wang. A comparative study of logic programs with preference. In B. Nebel, editor, *Proceedings of the seventeenth International Conference on Artificial Intelligence (IJCAI-01)*, pages 597–602, San Francisco, CA, Aug. 4–10 2001. Morgan Kaufmann Publishers, Inc.
- [45] T. Tsouanas. Semantic approaches to logic programming. Master’s thesis, University of Athens, 2010.
- [46] M. van Emden and R. Kowalski. The semantics of predicate logics as a programming language. *Journal of the ACM*, 23(4):733–742, 1976.
- [47] A. van Gelder, K. A. Ross, and J. S. Schilpf. The well-founded semantics for general logic programs. *Journal of the ACM*, 38(3):620–662, 1991.
- [48] G. H. von Wright. *The Logic of Preference, an Essay*. Edinburgh University Press, Edinburgh, 1963.
- [49] W. W. Wadge. Infinitesimal logic, Bill Wadge’s Blog, 2011.
- [50] K. Wang, L. Zhou, and F. Lin. Alternating fixpoint theory for logic programs with priority. *Lecture Notes in Computer Science*, 1861:164–??, 2000.
- [51] M. Wilson and A. Borning. *Hierarchical constraint logic programming*. Dept. Computer Science and Engineering, University of Washington, Seattle., 1991.
- [52] Y. Zhang and N. Y. Foo. Answer sets for prioritized logic programs. In *ILPS*, pages 69–83, 1997.
- [53] Π. Σταματόπουλος. *Λογικός και Συναρτησιακός Προγραμματισμός*. Ελληνικό Ανοικτό Πανεπιστήμιο (υπό έκδοση), 2003.