ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ
ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ & ΔΙΚΤΥΩΝ

Διπλωματική Εργασία

ΑΝΑΠΤΥΞΗ ΥΠΟΛΟΓΙΣΤΙΚΟΥ ΠΕΡΙΒΑΛΛΟΝΤΟΣ ΜΗ ΜΟΝΟΤΟΝΗΣ ΣΥΛΛΟΓΙΣΤΙΚΗΣ

Τελειόφοιτη Φοιτήτρια:
Κουμπουλή Παναγιώτα 1701052

Βόλος, Ιούνιος 2007

Εγκρίθηκε από τα μέλη της Διμερούς Εξεταστικής Επιτροπής

Πρώτος Εξεταστής (Επιβλέπων Καθηγητής)	Ασπασία Δασκαλοπούλου Επίκουρος Καθηγήτρια, Τμήμα Μηχανικών Η/Υ Τηλεπικοινωνιών & Δικτύων, Πανεπιστήμιο Θεσσαλίας
Δεύτερος Εξεταστής	Βασίλειος Βερύκιος Επίκουρος Καθηγητής , Τμήμα Μηχανικών Η/Υ Τηλεπικοινωνιών & Δικτύων, Πανεπιστήμιο Θεσσαλίας

Ευχαριστίες

Θα ήθελα να ευχαριστήσω, καταρχήν, την επιβλέποντα καθηγήτριά μου Επίκουρο Καθηγήτρια κ. Ασπασία Δασκαλοπούλου για τη συνεργασία μας και τη βοήθεια που μου προσέφερε για την εκπόνηση της διπλωματικής μου εργασίας, όπως επίσης και τον δεύτερο υπεύθυνο καθηγητή για τη διπλωματική μου εργασία, Επίκουρο Καθηγητή κ. Βασίλειο Βερύκιο καθώς και τον διδακτορικό φοιτητή Γεώργιο Γιαννίκη. Επίσης, είμαι ευγνώμων σε όλους τους καθηγητές μου, στο Τμήμα Μ.Η.Υ.Τ.Δ, για τη γνώση και τις διδαχές που μου προσέφεραν τα χρόνια που φοίτησα στο Πανεπιστήμιο.

Ένα μεγάλο ευχαριστώ στην οικογένεια μου, την μητέρα μου και την αδερφή μου, που είναι πάντα δίπλα μου, με στηρίζουν σε κάθε μου απόφαση και ένα μεγάλο μέρος όσων έχω καταφέρει μέχρι τώρα, το χρωστώ σε εκείνους. Επίσης, ευχαριστώ την Στεφανία Νιαβή, την Βάγια Μπούρα, τη Φανή Λιάκου και τον Νικόλαο Μήτση για την πολύτιμη βοήθειά τους και ηθική συμπαράσταση κατά την διάρκεια εκπόνησης της διπλωματικής μου. Τέλος, ευχαριστώ τους φίλους μου και τους ανθρώπους που βρίσκονται κοντά μου, γιατί υπάρχουν στη ζωή μου και είναι μαζί μου σε κάθε βήμα που κάνω μικρό ή μεγάλο.

ΠΕΡΙΕΧΟΜΕΝΑ

Abstract

Ο εύλογος συλλογισμός (default reasoning) είναι μία από τις πιο δημοφιλής προσεγγίσεις μη μονότονης συλλογιστικής. Παρέχει τη δυνατότητα χειρισμού καταστάσεων για τις οποίες έχουμε ελλιπής πληροφόρηση. Οι default κανόνες χρησιμοποιούνται σε ένα εύρος εφαρμογών όπως την ιατρική διαγνωστική και την νομική συλλογιστική. Σκοπός αυτής της διπλωματικής είναι η μελέτη διαφόρων θεωριών γύρω από τον εύλογο συλλογισμό και η υλοποίηση ενός αλγορίθμου με την χρήση λογικού προγραμματισμού και συστημάτων κανόνων για την εύρεση επεκτάσεων (extension) με έναν τρόπο παρόμοιο με αυτόν που πρότεινε ο Αντωνίου (Paper: A tutorial on Default Logics) τροποποιημένο όμως έτσι ώστε να είναι περισσότερο ευέλικτος και χρήσιμος και να μπορεί να συνδυαστεί με την λογική που βασίζεται σε επιχειρήματα που πρότεινε ο Prakken (Paper: Two Approaches to Formalization of Defeasible Deontic Logic).

<u>Κεφάλαιο 1ο</u>
Εισαγωγή

Η Τεχνητή Νοημοσύνη είναι ο κλάδος της επιστήμης των υπολογιστών που περιλαμβάνει νέες στρατηγικές έρευνας και μεθόδους αναπαράστασης της γνώσης σε προγράμματα ηλεκτρονικών υπολογιστών, στην προσπάθεια να εξομοιωθεί η διαδικασία που ακολουθεί το ανθρώπινο μυαλό για την επίλυση διαφόρων προβλημάτων ή για την λήψη κάποιων αποφάσεων. Από μια άποψη ο στόχος της τεχνητής νοημοσύνης είναι να κάνει τους υπολογιστές περισσότερο χρήσιμους για τους ανθρώπους. Ένας δεύτερος σκοπός της τεχνητής νοημοσύνης, αλλά εξίσου σημαντικός είναι η καλύτερη κατανόηση της ανθρώπινης νοημοσύνης. Η δημιουργία ενός *νοήμονος υπολογιστικού συστήματος* απαιτεί να καταλάβουμε πώς οι άνθρωποι αντιλαμβάνονται, συγκεντρώνουν, οργανώνουν και χρησιμοποιούν τη γνώση για την επίλυση ενός προβλήματος.

Ένας από τους πιο διαδεδομένους τομείς της Τεχνητής Νοημοσύνης είναι τα υπολογιστικά συστήματα λογικής που μπορούν να επεξεργάζονται δεδομένα (υπάρχουσα γνώση) και να μπορούν να εξάγουν συμπεράσματα, να παίρνουν αποφάσεις σε μια συγκεκριμένη στιγμή ή να προσαρμόζονται σε ορισμένες καταστάσεις αλλάζοντας την κατάστασή τους. Για να επιτευχθούν όμως αυτά θα πρέπει να υπάρχει ένας ευέλικτος και χρήσιμος τρόπος αναπαράστασης της γνώσης και των δεδομένων.

Αναπαράσταση Γνώσης και Συλλογιστικές.

Η αναπαράσταση γνώσης στην καθημερινή ζωή γίνεται συνήθως με απλή περιγραφή σε φυσική γλώσσα. Όμως, η φυσική γλώσσα θεωρείται ακατάλληλη για αναπαράσταση γνώσης σε υπολογιστικά συστήματα κυρίως λόγω της πολύ-σημαντικότητά της (επιδέχεται μια φράση πολλές ερμηνείες). Ιδιαίτερα για τα συστήματα τεχνητής νοημοσύνης απαιτείται ένας μονοσήμαντος και τυποποιημένος συμβολισμός, ο οποίος εκτός της δυνατότητας που θα δίνει για ακριβή αναπαράσταση της γνώσης, θα πρέπει να μπορεί να συνδυαστεί κατάλληλα με μία ή περισσότερες συλλογιστικές για την εξαγωγή συμπερασμάτων. Έτσι, ή αναπαράσταση γνώσης θεωρείται ένα σύνολο συντακτικών και σημασιολογικών παραδοχών, οι οποίες καθιστούν δυνατή την περιγραφή ενός κόσμου. Μια μέθοδος, λοιπόν, αναπαράστασης γνώσης έχει

- συντακτικό (syntax): ορισμός συμβόλων και κανόνες συνδυασμού τους
- σημασιολογία (semantics): καθορισμός εννοιών συμβόλων και
- ένα μηχανισμό εξαγωγής συμπερασμάτων (inference mechanism): τρόπος εξαγωγής συμπερασμάτων από υπάρχουσα γνώση.

Ο μηχανισμός εξαγωγής συμπερασμάτων υλοποιείται από τη *συλλογιστική (reasoning)*. Η συλλογιστική είναι η μέθοδος με την οποία τμήματα υπάρχουσας γνώσης συνδυάζονται μεταξύ τους για να παράγουν νέα γνώση ή να εξάγουν συμπεράσματα. Οι πιο διαδεδομένες συλλογιστικές είναι η *συν-επαγωγική συλλογιστική (deductive reasoning)*, η επαγωγική *συλλογιστική (inductive reasoning)* και η *απαγωγική συλλογιστική (abductive reasoning)*. Η συν-επαγωγική συλλογιστική από ένα σύνολο γεγονότων εξάγει συμπεράσματα βασισμένη στους κλασικούς μηχανισμούς εξαγωγής συμπερασμάτων της λογικής. Η επαγωγική, αφορά την εξαγωγή γενικών συμπερασμάτων από ένα σύνολο παραδειγμάτων. Ενώ, η απαγωγική συλλογιστική από τα δεδομένα μιας βάσης γνώσης και μερικές παρατηρήσεις επιχειρεί την εύρεση υποθέσεων οι οποίες μαζί με τη ισχύουσα γνώση εξηγούν τις παρατηρήσεις.

Υπάρχουν τέσσερα είδη γνώσης (Bar και Feigenbaum), τα οποία πρέπει να είναι ικανή να αναπαραστήσει κάθε μέθοδος αναπαράστασης γνώσης: 1.τα *αντικείμενα (objects)*. Αναπαράσταση των αντικειμένων ενός κόσμου, της κλάσης στην οποία ανήκουν καθώς και της σχέσης μεταξύ των κλάσεων αυτών. (Σημασιολογική Γνώση). 2. τα *γεγονότα (facts)*. Αναπαράσταση των ενεργειών ή περιστατικών που συμβαίνουν σε ένα κόσμο, της χρονικής ακολουθίας με την οποία γίνονται, καθώς και τις σχέσεις αίτιου – αποτελέσματος. (Επεισοδιακή Γνώση). 3.η *εκτέλεση (performance)*. Αναπαράσταση του τρόπου με τον οποίο εκτελείται μια εργασία ή διεκπεραιώνεται μια διαδικασία. (Διαδικαστική Γνώση). 4. Μετά – γνώση (meta – knowledge). Γνώση για το πώς και πότε μπορεί να χρησιμοποιηθεί κάθε τμήμα γνώσης. (Σοφία).

Τα κριτήρια αξιολόγησης (Rich) μια μεθόδου αναπαράστασης γνώσης είναι:

- ✓ *Επάρκεια αναπαράστασης (representational adequacy)*: Η ικανότητα να αναπαριστά όλα τα είδη γνώσης.
- ✓ *Επάρκεια συνεπαγωγής (inferential adequacy)*: Η ικανότητα να συνεργάζεται με μηχανισμούς επεξεργασίας υπαρχουσών δομών γνώσης με σκοπό τη δημιουργία νέας γνώσης.
- ✓ *Αποδοτικότητα συνεπαγωγής (inferential efficiency)*: Η ικανότητα να εισάγει επιπλέον πληροφορίες στις δομές γνώσης, ώστε να κατευθύνει τους μηχανισμούς εξαγωγής συμπερασμάτων ταχύτερα προς τη λύση.
- ✓ *Αποδοτικότητα απόκτησης (acquisitional efficiency)*: Η ικανότητα να επιτρέπει την απόκτηση νέας γνώσης εύκολα και γρήγορα.

Λογική.

Μια διαδεδομένη μέθοδος αναπαράστασης γνώσης αποτελεί η Λογική. Η λογική παρέχει έναν τρόπο για την αποσαφήνιση και την τυποποίηση της ανθρώπινης σκέψης και προσφέρει μια εύχρηστη και σημαντική μεθοδολογία για την αναπαράσταση και την επίλυση προβλημάτων. Έχει υψηλό βαθμό εκφραστικότητας και οποιαδήποτε δεδομένα, γεγονότα, αντιλήψεις, αντικείμενα και σχέσεις που αφορούν ένα κόσμο μπορούν να δηλωθούν και να επεξεργαστούν με εύκολο και ευέλικτο τρόπο. Επιπλέον, η λογική πρώτης τάξης μπορεί να συνδυαστεί με τους default κανόνες και να μπορούμε να αναπαριστούμε τόσο τα βέβαια γεγονότα (facts) που γνωρίζουμε από τη βάση γνώσης ότι ισχύουν όσο και κάποιες πεποιθήσεις που εξάγονται μετά την εφαρμογή των default κανόνων και δεν είμαστε βέβαιοι ότι ισχύουν αλλά δεν μας εμποδίζει τα και τίποτα (κάποιο γεγονός) να υποθέσουμε την ισχύ τους.

Σε πολλές εφαρμογές, όπως στους πράκτορες, στα έμπειρα συστήματα και τα νευρωνικά δίκτυα πίσω από τον προγραμματισμό τους κρύβεται μια μορφή λογικής. Στους τομείς της ιατρικής, της νομικής, στην υπηρεσία πρόβλεψης του καιρού χρησιμοποιείται μια μορφή εύλογου συλλογισμού(default reasoning) για να μπορούν να προβλεφθούν κάποια δεδομένα ή να παρθούν αποφάσεις. Γενικότερα, όταν υπάρχει ελλιπής πληροφόρηση ή γνώση γύρω από έναν κόσμο ή μια κατάσταση και ωστόσο πρέπει να παρθούν κάποιες αποφάσεις ή να γίνουν κάποιες αλλαγές χωρίς να υπάρχει η πολυτέλεια χρόνου ώστε να συλλεχθούν επιπλέον χρήσιμα δεδομένα χρησιμοποιούνται εύλογοι συλλογισμοί.

Ωστόσο, επειδή τα συμπεράσματα στα οποία καταλήγουμε με τον εύλογο συλλογισμό (default reasoning) μερικές φορές αποδεικνύεται στο μέλλον μετά τη συλλογή περαιτέρω δεδομένων και άρα γνώσης ότι δεν έπρεπε να δεχτούμε ότι ισχύουν καθώς είναι εσφαλμένα θα πρέπει να υπάρχει ένας τρόπος ή να γυρίσουμε στην προηγούμενη κατάσταση όπου δεν ίσχυαν έτσι όμως χάνουμε ότι νέα πληροφορία πήραμε μετά από αυτήν την κατάσταση και η οποία δεν βασίστηκε στα στοιχεία του default κανόνα καθώς και χρόνο ή να μπορέσουμε να τα αναιρέσουμε μόνο αυτά και ότι σχετίζεται με αυτά χωρίς να πρέπει να πάμε πίσω απλά να μεταβούμε σε μία νέα κατάσταση με λιγότερα δεδομένα Ο μοναδικός τρόπος επίτευξης αυτού έτσι ώστε να εξασφαλίζεται η συνέπεια και η ορθότητα της νέας γνώσης είναι με τη χρήση μη μονότονης λογικής.

Μη Μονότονη Λογική.

Σε μια μονότονη λογική, όπως για παράδειγμα είναι ο προτασιακός λογισμός (Παράρτημα), υπάρχει ένα σύστημα αξιωμάτων S (η αρχική βάση γνώσης) και ένα σύνολο τύπων F που αποδεικνύονται από το S με κάποιο μηχανισμό εξαγωγής συμπερασμάτων. Η προσθήκη ενός ή περισσοτέρων αξιωμάτων στο S (απόκτηση νέας γνώσης), στη χειρότερη θα διατηρήσει το σύνολο F σταθερό και συνήθως το πιθανότερο είναι να το αυξήσει. Με άλλα λόγια, αν ένα συμπέρασμα μπορεί να συναχθεί από ένα σύνολο προτάσεων λογισμού S τότε μπορεί να συναχθεί και από ένα μεγαλύτερο σύνολο που υπερσύνολο του S. Η αλήθεια των αρχικών υποθέσεων εγγυάται την αλήθεια των εξαγόμενων συμπερασμάτων και η προσθήκη νέας γνώσης δεν μπορεί να την υπονομεύσει. Δηλαδή όταν ο αριθμός των αξιωμάτων αυξάνεται, το σύνολο F αυξάνει και αυτό μονότονα.

Τα πλεονεκτήματα αυτής της λογικής που οδηγούν στην εκτεταμένη χρήση της είναι:

- Κάθε φορά που προστίθεται ένα νέο γεγονός στο *S*, δε χρειάζονται νέοι έλεγχοι για τη συνέπεια της γνώσης του συστήματος
- Για κάθε νέο γεγονός που αποδεικνύεται δεν είναι απαραίτητη η καταγραφή των γεγονότων πάνω στα οποία βασίζεται η αλήθεια του, αφού δεν υπάρχει κίνδυνος απομάκρυνσης παλαιότερων γεγονότων.

Αντίθετα σε μια μη μονότονη λογική η προσθήκη νέων αξιωμάτων είναι δυνατό να μειώσει το σύνολο των συμπερασμάτων, αφαιρώντας κάποια που αποδεικνύονται εσφαλμένα μετά την αύξηση του συνόλου S.

Οι μη- μονότονες συλλογιστικές είναι κατάλληλες για την αντιμετώπιση κάποιων καταστάσεων που εμφανίζονται συχνά στον πραγματικό κόσμο:

- Καταστάσεις για τις οποίες δεν έχουμε πλήρη γνώση, ή η γνώση δημιουργείται κατά τη διάρκεια της εκτέλεσης ενεργειών, για τις οποίες δεν είμαστε βέβαιοι για την αναγκαιότητα ή ορθότητά τους.
- Καταστάσεις στις οποίες η γνώση μεταβάλλεται, λόγω μεταβολών που συμβαίνουν στον κόσμο.
- Καταστάσεις στις οποίες το σύστημα χρησιμοποιεί υποθέσεις (assumptions) στα πλαίσια της στρατηγικής επίλυσης προβλημάτων.

Μια περίπτωση μη μονότονης λογικής είναι η *μη-μονότονη τροπική λογική* (*non-monotonic modal logic*) στην οποία εισάγεται ένας νέος *τροπικός τελεστής* ο οποίος δηλώνει ότι ένα γεγονός "είναι συνεπές με τις τρέχουσες πεποιθήσεις". Η λογική αυτή εισάγει έναν μηχανισμό εξαγωγής συμπερασμάτων που σε συνδυασμό με τον τροπικό τελεστή επιτρέπει τη διατήρηση της συνέπειας της γνώσης κατά την κατάργηση γεγονότων.

Μια άλλη μη μονότονη συλλογιστική είναι η *συλλογιστική εύλογων υποθέσεων* (default reasoning). Η συλλογιστική αυτή χρησιμοποιείται σε περιπτώσεις κατά τις οποίες ένα γεγονός συνάγεται από ένα δοσμένο γεγονός, γιατί έτσι συμβαίνει συνήθως και γιατί δεν υπάρχει ένδειξη για το αντίθετο. Ωστόσο, αν κάποια στιγμή αργότερα εμφανιστούν ενδείξεις για την μη ορθότητα κάποιου γεγονότος, το γεγονός αυτό αποσύρεται. Αυτή η λογική χρησιμοποιεί τη λογική πρώτης τάξης (Παράρτημα) επαυξημένη με μια νέα μορφή κανόνων που ονομάζονται *default rules* και έχουν την μορφή:

$$\frac{p\ :\ q}{r}$$

όπου το p αποτελεί την προϋπόθεση (prerequisite) , το q τη δικαιολογία (justification) και το r το επακόλουθο (consequent). Αυτοί οι κανόνες διαβάζονται: "Αν ισχύει p και είναι συνεπές με την βάση γνώσης να υποθέσουμε το q, τότε μπορούμε να συμπεράνουμε το r".

Μια απλή κατηγορία default κανόνων είναι τα normal defaults στα οποία τα q και r είναι τα ίδια. Τότε ο κανόνας μεταφράζεται ως εξής: Αν αποδεικνύεται η προϋπόθεση τότε μπορούμε να εξάγουμε το συμπέρασμα αν είναι συνεπές με την υπόλοιπη βάση γνώσης.

Παράδειγμα:

Ο κανόνας ότι τα πουλιά τυπικά πετάνε μπορεί να αναπαρασταθεί ως εξής:

$$\frac{Bird\ (X)\ :\ Flies\ (X)}{Flies\ (X)}$$

Αιτία και στόχος διπλωματικής.

Καθώς ο εύλογος συλλογισμός (default reasoning) είναι ένα από τους πιο ενδιαφέρων και νέο-αναπτυσσόμενος τομέας της τεχνητής νοημοσύνης με πληθώρα εφαρμογών η ενασχόληση και η μελέτη των προσεγγίσεών του είναι ελκυστική. Μετά την ασχολία με την θεωρία του Reiter(Paper: A logic for default reasoning (1980)), τον αλγόριθμο του Αντωνίου (Paper: A tutorial on Default Logics) για την εύρεση extensions και το πλαίσιο εργασίας του Prakken (Paper: Two Approaches to Formalization of Defeasible Deontic Logic) ασχολήθηκα με την υλοποίηση ενός αλγορίθμου που επεκτείνει τον αλγόριθμο του Αντωνίου και τον συνδυάζει με την θεωρία του συλλογισμού με επιχειρήματα του Prakken. Πιο συγκεκριμένα, υιοθετείται μια δικαιολογημένη default θεωρία για την εύρεση extension σε συνδυασμό με τον αλγόριθμο του Αντωνίου. Δηλαδή, δεν απαιτείται να εφαρμόζουμε όλα τους κανόνες που μπορούν να εφαρμοστούν αν αυτό μας οδηγεί σε εσφαλμένη νέα γνώση. Το κύριο μέλημα μας είναι να επεκτείνουμε τη βάση γνώσης μας διασφαλίζοντας μόνο την συνέπεια των συμπερασμάτων και των προϋποθέσεων που στηρίχτηκε η εξαγωγή τους. Στην συνέχεια αυτές οι επεκτάσεις στηρίζονται σε διαδικασίες που ουσιαστικά είναι επιχειρήματα που κατηγοριοποιούνται σε δικαιολογημένα (justified), ανατρεπόμενα (overruled) και αναιρέσιμα (defeasible) όπως τα ορίζει και ο Prakken. (Αναιρέσιμη Λογική: Παράρτημα).

Ο στόχος του αλγορίθμου αυτού είναι να μπορούμε να παράγουμε όσον το δυνατόν περισσότερη νέα γνώση (περισσότερα συμπεράσματα) ή να λαβαίνουμε γρηγορότερα αποφάσεις χωρίς όμως να χάνουμε την συνέπεια της βάσης γνώσης που στηρίζεται το υπολογιστικό σύστημα ή να έχουμε αντιφάσεις και αντικρουόμενα συμπεράσματα.

Στο πρώτο κεφάλαιο της διπλωματικής, λοιπόν, παρουσιάστηκαν οι λόγοι, η ιδέα, ο τρόπος και ο στόχος της έρευνας αυτής. Στη συνέχεια, εξηγούνται και παρατίθενται βασικές έννοιες της θεωρίας του Reiter η οποία είναι και το θεωρητικό υπόβαθρο του εύλογου συλλογισμού. Ακόμα, περιέχεται η θεωρία και ο αλγόριθμος του Αντωνίου στον οποίο βασίστηκε και ο αλγόριθμος που υλοποιήθηκε. Επεξηγείται το πλαίσιο εργασίας του Prakken, έπειτα παρουσιάζεται η ιδέα και ο αλγόριθμος και κάποια παραδείγματα που αναδεικνύουν την χρησιμότητα του και τον τρόπο λειτουργίας του και καταλήγει στην καταγραφή των δυσκολιών κατά τη διάρκεια διεξαγωγής της υλοποίησης, στις επεκτάσεις και στους μελλοντικούς στόχους της εργασίας αυτής.

Κεφάλαιο 2ο
Θεωρία του Reiter.

Εισαγωγή στον Εύλογο Συλλογισμό (Default Reasoning).

Ο εύλογος συλλογισμός (Default Reasoning) είναι η διαδικασία εξαγωγής συμπερασμάτων που βασίζονται σε σχέδια συμπερασμού της μορφής 'με την απουσία κάποιας πληροφορίας για το αντίθετο μπορούμε να υποθέσουμε ότι ...'. Αυτού του είδους τα πρότυπα συλλογισμού αντιπροσωπεύουν μια μορφή συλλογισμού που οδηγεί στην εξαγωγή συμπερασμάτων που πιθανόν να είναι έγκυρα και χρησιμοποιούνται σε περιπτώσεις που υπάρχει ελλιπής γνώση σχετικά με ένα κόσμο. Τα συμπεράσματα που βασίζονται πάνω στην default συλλογιστική αντιμετωπίζονται ως παροδικές αντιλήψεις που μπορεί να τροποποιηθούν, να ισχυροποιηθούν ή να καταργηθούν από τις επόμενες παρατηρήσεις και τις νέες πληροφορίες.

Παράδειγμα εύλογου συλλογισμού:

Μερικές φορές είναι αναγκαίο να καταλήξουμε σε ένα συμπέρασμα χωρίς να έχουμε πλήρη γνώση του κόσμου του προβλήματος. Πολύ συχνές είναι η διατυπώσεις της μορφής: 'Τα περισσότερα x είναι y εκτός από ...'. Για να αναπαρασταθεί αυτή η μορφή γνώσης χρησιμοποιούνται οι default κανόνες σε συνδυασμό με κάποιους κανόνες εξαιρέσεων. Για παράδειγμα, έστω ότι στη βάση γνώσης έχουμε τις εξής πληροφορίες:

1. Τα περισσότερα πουλιά πετάνε.
2. Ο πιγκουίνος και η στρουθοκάμηλος παρόλο που είναι πουλιά δεν πετάνε.

Για να εκφραστούν αυτά τα δεδομένα θα πρέπει με κάποιον τρόπο να αναπαρασταθεί το εξής:

'Αν το x είναι πουλί, τότε με την απουσία κάποιας πληροφορίας για το αντίθετο είναι δυνατό να εξαχθεί το συμπέρασμα ότι το x πετάει'.

Το δύσκολο είναι να δηλωθεί η φράση 'με την απουσία κάποιας πληροφορίας για το αντίθετο'. Ο Reiter υιοθέτησε την μετάφραση 'Είναι συνεπές να υποθέσουμε ότι το x μπορεί να πετάξει'. Έτσι, η προηγούμενη έκφραση μετατράπηκε στην:

'Αν το x είναι πουλί και είναι συνεπές να υποθέσουμε ότι πετάει τότε καταλήγουμε στο συμπέρασμα ότι πετάει'.

Και ο τρόπος αναπαράστασης μέσω ενός default κανόνα είναι:

$$\frac{\text{ΠΟΥΛΙ}(x) : \text{ΜΠΕΤΑΕΙ}(x)}{\text{ΠΕΤΑΕΙ}(x)}$$

όπου το Μ μεταφράζεται ως 'είναι συνεπές να υποθέσουμε'.

Οι εξαιρέσεις μπορούν να αναπαρασταθούν με τη βοήθεια της standard λογική πρώτης τάξης (Παράρτημα) ως:

$$(x).\text{ΠΙΓΚΟΥΙΝΟΣ}(x) \supset \neg\,\text{ΠΕΤΑΕΙ}(x)$$
$$(x).\text{ΣΤΡΟΥΘΟΚΑΜΗΛΟΣ}(x) \supset \neg\,\text{ΠΕΤΑΕΙ}(x)$$

Θα πρέπει να σημειωθεί ότι τα συμπεράσματα που εξάγονται από default κανόνες αντιμετωπίζονται ως πιστεύω και υπόκεινται σε πιθανή τροποποίηση, κατάργηση ή ισχυροποίηση ανάλογα με τις νέες πληροφορίες που θα προκύψουν μελλοντικά.

Υπόθεση Κλειστού Κόσμου (Closed World Assumption):

Εάν η R είναι μια n-οστή σχέση, τότε ο παρακάτω είναι ένας κλειστού κόσμου κανόνας:

$$\frac{: M\neg R(x_1,\ldots,x_n)}{: \neg R(x_1,\ldots,x_n)}$$

Αν αυτός ο default κανόνας είναι σε ισχύ για όλες τις σχέσεις R μιας περιοχής ενδιαφέροντος, τότε ο συλλογισμός γίνεται με την υπόθεση κλειστού κόσμου. Ουσιαστικά, με την υπόθεση κλειστού κόσμου για κάθε σχέση R και για κάθε ανεξάρτητες μεταβλητές $x_1,\ldots,x_n$ μπορεί να υποτεθεί ότι ισχύει $\neg R(x_1,\ldots,x_n)$ όποτε είναι αυτό συνεπές.

Για παράδειγμα, σε μια βάση δεδομένων που αντιπροσωπεύει ένα πρόγραμμα αεροπορικών πτήσεων, για να απαντηθεί η ερώτηση αν μια πτήση συνδέει δυο πόλεις μεταξύ τους δεν χρειάζεται να καταγραφούν αναλυτικά για κάθε πτήση ποιες πόλεις δεν συνδέονται μεταξύ τους. Η πληροφορία για κάθε πτήση ποιες πόλεις συνδέει είναι αρκετή για να συμπεράνουμε με συνέπεια ότι μέσω μιας πτήσης δυο πόλεις δεν συνδέονται όταν δεν έχουμε κάποια πληροφορία για την σύνδεση τους.

Με αυτόν τον τρόπο, η αναπαράσταση του κόσμου απλουστεύεται και επιτυγχάνεται υπολογιστικό κέρδος. Πραγματικά, μόνο θετικές πληροφορίες χρειάζεται να καταγράφονται αφού οι αρνητικές μπορούν να εξάγονται μέσω των defaults.

Μη μονοτονικότητα του εύλογου συλλογισμού:

Επειδή τα συμπεράσματα που εξάγονται μέσω ενός default μπορεί με την προσθήκη νέας γνώσης για τον κόσμο να αναιρεθούν ή τροποποιηθούν η συλλογιστική με default κανόνες είναι μη μονότονη. Αυτό οδηγεί στην ανάγκη όμως μαζί με κάθε αντίληψη που σχηματίστηκε από κάποιο default να συγκρατούνται και τα δεδομένα στα οποία στηρίχτηκε η αποδοχή της. Κι αυτό γιατί αν κάποιο δεδομένο από αυτά μετά από περαιτέρω μελλοντική παρατήρηση παύσει να ισχύει τότε θα πρέπει και το πιστεύω να απορριφτεί.

Defaults και Ατελής Γνώση:

Στην προσπάθεια να περιγραφεί πλήρως κάθε περίπλοκος κόσμος με την χρήση λογικής πρώτης τάξης επειδή υπάρχουν πολλά κενά στην βάση γνώσης για αυτόν η θεωρία κατηγορηματικής λογικής είναι ελλιπής. Κάποιες φορές είναι αναγκαίο να παρθούν αποφάσεις ή να εξαχθούν συμπεράσματα παρά την μη πληρότητα πληροφοριών της βάσης γνώσης. Εκεί έγκειται ο ρόλος ενός default κανόνα να συμπληρώσει τις ελλείψεις των δεδομένων και κατ' επέκταση να εμπλουτίσει τις δυνατότητες της λογικής πρώτης τάξης. Οι default κανόνες, λοιπόν, λειτουργούν ως κάποια μορφή μετά-κανόνων, είναι οδηγίες σχετικά με το πώς θα δημιουργηθεί μια extension (επέκταση) της ελλιπής θεωρίας. Ωστόσο, υπάρχουν πολλοί τρόποι επέκτασης μια μη πλήρης θεωρίας για αυτό και οι default κανόνες μπορεί να είναι μη ντετερμινιστικοί. Διαφορετικές εφαρμογές των defaults οδηγούν σε διαφορετικές επεκτάσεις και ουσιαστικά σε διαφορετικές αντιλήψεις για τον κόσμο.

Ως παράδειγμα, έστω ότι ισχύει η υπόθεση κλειστού κόσμου. Η θεωρία $p \vee q$ μαζί με τους default κανόνες κλειστού κόσμου $:M\neg p / \neg p$ και $:M\neg q / \neg q$ έχει δύο διαφορετικές πλήρεις επεκτάσεις, $\{p, \neg q\}$ και $\{\neg p, q\}$. Στην πρώτη επιλέξαμε να πιστέψουμε $\neg q$ ενεργοποιώντας τον δεύτερο κανόνα, και έτσι αναγκαστήκαμε να πιστέψουμε λόγω της θεωρίας και p. Στην δεύτερη επιλέξαμε να πιστέψουμε $\neg p$ από την πυροδότηση του πρώτου κανόνα και q.

Θεωρίες με defaults και extensions (επεκτάσεις).

Οι extensions επίσημα καθορίζουν το σύνολο των αντιλήψεων που δημιουργούνται από μία ομάδα default κανόνων, ως ένα τρόπο περαιτέρω συμπλήρωσης κάποιας ισχύουσας ελλιπής συλλογής από γεγονότα σχετικά με έναν κόσμο.

Προπαρασκευαστική γνώση:

Ως L συμβολίζουμε το σύνολο των καλά ορισμένων τύπων (wff) λογικής πρώτης τάξης το οποίο σχηματίζεται χρησιμοποιώντας ένα αλφάβητο A που περιλαμβάνει πολλές αλλά υπολογίσιμες: μεταβλητές x, y, z … , γράμματα συναρτήσεων a, b, c …, κατηγορήματα P, Q, R … καθώς και τα σύμβολα στίξης, τις σταθερές της standard λογικής $\neg(not), \wedge(and), \vee(or), \supset (implies)$ και τους ποσοδείκτες (x) (για όλα τα x), (Ex) (υπάρχει ένα x).

Ένας καλά ορισμένος τύπος είναι κλειστός αν και μόνο αν δεν περιλαμβάνει ελεύθερες μεταβλητές. Για κάθε σύνολο κλειστών καλά ορισμένων τύπων S, και για κάθε καλά ορισμένο τύπο w, S |-- w σημαίνει ότι το w είναι δυνατόν να αποδειχθεί με τη λογική πρώτης τάξης από τις συμβάσεις του S. Για κάθε σύνολο λοιπόν S υποσύνολο του L ορίζεται η Th(S) = { $w \mid w \in L,$όπου το w είναι κλειστός καλά ορισμένος τύπος και S |-- w}.

Ένα default είναι κάθε έκφραση της μορφής:

$$\frac{\alpha(x) : M\beta_1(x),\dots,M\beta_m(x)}{w(x)}$$

όπου α(x), β_1(x), …, β_m(x), w(x) είναι καλά ορισμένοι τύποι των οποίων οι ελεύθερες μεταβλητές είναι ανάμεσα σε αυτές του x = x_1, …, x_n και το α(x) ονομάζεται προαπαιτούμενο (prerequisite) του default και το w(x) επακόλουθο (consequent). Ένα default ονομάζεται κλειστό όταν τα α, β_1, …, β_m, w περιλαμβάνουν μία ελεύθερη μεταβλητή.

Μια θεωρία με defaults ορίζεται ως ένα ζευγάρι (D,W) όπου το D είναι το σύνολο των defaults κανόνων και το W το σύνολο των καλά ορισμένων τύπων(wffs). Μια default θεωρία (D,W) είναι κλειστή αν και μόνο αν κάθε default του D είναι κλειστό.

Οι επεκτάσεις μιας κλειστού κόσμου default θεωρίας:

Ορισμός: Έστω ότι η Δ=(D,W) είναι μια default θεωρία κλειστού κόσμου, έτσι ώστε κάθε κανόνας default του D είναι της μορφής $\frac{\alpha : M\beta_1,\dots,M\beta_m}{w}$ όπου α, β_1, …, β_m, w είναι κλειστού κόσμου καλά ορισμένοι τύποι του L. Για κάθε σύνολο από κλειστούς καλά ορισμένους τύπους $S \subseteq L$ έστω ότι το Γ(S) είναι το μικρότερο σύνολο που ικανοποιεί της παρακάτω τρεις ιδιότητες:

1. $W \subseteq \Gamma(S)$
2. $Th_L(\Gamma(S)) = \Gamma(S)$
3. Αν ο $(\frac{\alpha : M\beta_1,\dots,M\beta_m}{w}) \in D$ και $\alpha \in \Gamma(S)$ και $\neg\beta_1,\dots,\neg\beta_m \notin S$ τότε $w \in \Gamma(S)$.

Ένα σύνολο από κλειστούς καλά ορισμένους τύπους $E \subseteq L$ είναι μια extension για τη Δ αν και μόνο αν Γ(E) = E.

Θεώρημα: Έστω $E \subseteq L$ είναι ένα σύνολο από κλειστούς καλά ορισμένους τύπους, και έστω ότι η Δ=(D,W) είναι μια default θεωρία κλειστού κόσμου. Ορίζουμε:

$$E_0 = W$$

και για κάθε $i \geq 0$

$$E_{i+1} = Th(E_i) \cup \{w \mid \frac{a : M\beta_1,\dots,M\beta_m}{w} \in D, \text{όπου } \alpha \in E_i \text{ και } \neg\beta_1,\dots,\neg\beta_m \notin E\}.$$

Τότε το E είναι μια extension του Δ αν και μόνο αν:

$$E = \bigcup_{i=0}^{\infty} E_i.$$

Παράδειγμα: Έστω η θεωρία κλειστού κόσμου Δ με:

$$D = \{\frac{: MA}{A}, \frac{: MB}{B}, \frac{: MC}{C}\} \text{ και } W = \{B \supset \neg A \wedge \neg C\}.$$

Έχει δύο extensions (επεκτάσεις) τις:

$$E_1 = Th(W \cup \{A, C\}) \text{ και } E_2 = Th(W \cup \{B\}).$$

Παράδειγμα: Έστω η θεωρία κλειστού κόσμου Δ με:

$$D=\{\frac{A:M(Ex)P(x)}{(Ex)P(x)},\frac{:M\neg A:MA}{\neg A\quad A}\}\text{ και } W=\{\}.$$

Έχει δύο extensions (επεκτάσεις) τις:

$$E_1=Th(\{\neg A\})\text{ και } E_2=Th(\{A,(Ex)P(x)\}).$$

Αυτό το παράδειγμα μας δείχνει ότι υπάρχουν διαφορετικές επεκτάσεις της ίδιας default θεωρίας που υποστηρίζουν διαφορετικές και μπορεί και αντικρουόμενες οντότητες.

Επιπλέον, υπάρχουν και θεωρίες με defaults που δεν έχουν καμία επέκταση (extension).

Παράδειγμα: Έστω η θεωρία κλειστού κόσμου Δ με:

$$D=\{\frac{:MA}{\neg A}\}\text{ και } W=\{\}.$$

Αυτή δεν έχει καμία extension (επέκταση).

Σύνολο Παραγόμενων defaults: Αν η Δ = (D, W) είναι μια κλειστού κόσμου default θεωρία και η Ε είναι μια extension της Δ τότε το σύνολο των παραγόμενων defaults της Ε με σεβασμό στην Δ ορίζεται ως εξής:

$$GD(E,\Delta)=\{\frac{\alpha:M\beta_1,\ldots,M\beta_m}{w}\in D\mid a\in E\text{ και }\neg\beta_1,\ldots,\neg\beta_m\notin E\}.$$

Επακόλουθα ενός συνόλου D από defaults: Αν το D είναι ένα σύνολο από defaults (όχι απαραίτητα κλειστά) τότε τα επακόλουθα του D είναι:

$$CONSEQUENTS(D)=\{w(x)\mid\frac{\alpha(x):M\beta_1(x),\ldots,M\beta_m(x)}{w(x)}\in D\}.$$

Θεώρημα: Αν η Ε είναι μια extension μιας default θεωρίας (D, W) κλειστού κόσμου, και ισχύει ότι $B\subseteq E$ τότε η Ε είναι και μια extension της $(D,W\cup B)$.

Κανονικές Default Θεωρίες(Normal Default Theories).

Ένας default κανόνας λέγεται normal όταν έχει τη μορφή:

$$\frac{\alpha(x):Mw(x)}{w(x)}$$

δηλαδή όταν οι ισχυρισμοί που δικαιολογούν την ισχύ του (justifications) συμπίπτουν με τα επακόλουθα (conclusions) της εφαρμογή του.

Μια default θεωρία λοιπόν χαρακτηρίζεται κανονική (normal) αν και μόνο αν κάθε default κανόνας της έχει κανονική μορφή (normal).

Κανονικές default θεωρίες κλειστού κόσμου:

Θεώρημα: Κάθε κανονική default θεωρία κλειστού κόσμου έχει μία extension (επέκταση).

Ορισμός Ημι-Μονοτονικότητας (Semi-Monotonicity):

Θεώρημα: Έστω ότι τα D και D' είναι σύνολα από κλειστά normal defaults και ισχύει $D'\subseteq D$. Επίσης έστω ότι η Ε' είναι μια extension της κλειστού κόσμου normal default θεωρία Δ' = (D', W) και υπάρχει και η Δ = (D,W) θεωρία κλειστού κόσμου και κανονική. Τότε η Δ έχει μια extension Ε για την οποία ισχύει:

1. $E'\subseteq E$ και
2. $GD(E',\Delta')\subseteq GD(E,\Delta)$.

Θεώρημα: (Ορθογωνικότητα των Extensions) Αν μια κλειστού κόσμου κανονική default θεωρία (D, W) έχει δύο ξεχωριστές extensions E και F τότε η $E \cup F$ είναι μη συνεπής.

Πόρισμα: Αν η Δ =(D, W) είναι μια κανονική κλειστού κόσμου default θεωρία τέτοια ώστε το σύνολο $W \cup CONSEQUENTS(D)$ να είναι συνεπές τότε αυτή η θεωρία έχει μία και μοναδική extension.

Θεώρημα: Αν σε μια κλειστού κόσμου κανονική default θεωρία Δ = (D, W) ισχύει ότι $D' \subseteq D$ και επιπλέον οι E_1 και E_2 είναι διαφορετικές και ευδιάκριτες extensions της (D', W) τότε υπάρχουν δύο ξεχωριστές extensions E_1 και E_2 της Δ τέτοιες ώστε $E_1^{'} \subseteq E_1$ και $E_2^{'} \subseteq E_2$.

Το παραπάνω θεώρημα ουσιαστικά δηλώνει ότι ο αριθμός των extensions μιας κανονικής κλειστού κόσμου default θεωρίας είναι μονότονος.

Προαπαιτούμενα ενός συνόλου D από defaults: Με το PREREQUISITES(D) συμβολίζεται τα προαπαιτούμενα ενός συνόλου από defaults D και είναι ίσο με την συνένωση ($\wedge$) όλων των προαπαιτούμενων των default κανόνων που περιλαμβάνει το D.

Λήμμα: Αν η (D, W) είναι μια κανονική κλειστού κόσμου default θεωρία για την οποία ισχύει $(\beta : Mu / u) \in D$, W |-- β και το $W \cup (u)$ είναι συνεπές, τότε κάθε extension της default θεωρίας $(D, W \cup \{u\})$ είναι και extension της (D, W).

Πόρισμα: Αν η (D, W) είναι μια κανονική κλειστού κόσμου default θεωρία για την οποία ισχύει ότι $D' \subseteq D$, το $W \cup CONSEQUENTS(D')$ είναι συνεπές και W |-- PREREQUISITES(D'), τότε κάθε extension της $(D, W \cup CONSEQUENTS(D'))$ default θεωρίας είναι και extension της (D, W).

Αυθαίρετες Θεωρίες Εύλογου Συλλογισμού(Arbitrary Default Theories).

Ένα *ελεύθερο default* (open) έχει τη μορφή $\alpha(x){:}M\beta_1(x),\ldots,M\beta_m(x)/w(x)$ όπου τουλάχιστον ένα από τα α(x), $\beta_1(x),\ldots,\beta_m(x)$,w(x) περιέχουν ελεύθερες μεταβλητές του x. Η μετάφραση του θα είναι: "Για όλες της ανεξάρτητες μεταβλητές $x_1,\ldots,x_n$ αν το α(x) ισχύει και κάθε ένα από τα $\beta_1(x),\ldots,\beta_m(x)$ μπορούν να γίνουν πιστευτά με συνέπεια, τότε κάποιος επιτρέπεται να πιστέψει ότι ισχύει και το w(x)."

Θα πρέπει όμως να οριστούν ποιοι μπορεί να είναι οι ανεξάρτητοι τύποι του x. Έτσι αν μια default θεωρία ορίζεται μέσω μιας γλώσσας L_A της οποίας το αλφάβητο είναι το A και το σύνολο $F \subseteq A$ περιέχει όλα τα γράμματα των συναρτήσεων του A (συμπεριλαμβανομένου και τις συναρτήσεις μηδενικής τάξης και τα γράμματα των σταθερών) τότε κάθε όρος που κατασκευάζεται χρησιμοποιώντας γράμματα του F θεωρείται ένας ανεξάρτητος τύπος της default θεωρίας. Με αυτόν τον τρόπο, λοιπόν, οι ελεύθερες μεταβλητές ενός ανοιχτού default κυμαίνονται ανάμεσα σε αυτούς τους όρους (*άμεσα οριζόμενοι ανεξάρτητοι όροι*).

Ωστόσο, υπάρχουν και οι *έμμεσα οριζόμενοι ανεξάρτητοι τύποι* που δημιουργούνται από τον υπαρξιακό ποσοδείκτη. Ένας τρόπος χειρισμού αυτών των όρων είναι η χρήση των συναρτήσεων Skolem.

Skolemization:

Έστω ότι το w είναι ένας καλά ορισμένος τύπος του L_A. Χωρίς έλλειψη της γενικότητας έστω ότι το w είναι σε μορφή λογικής πρώτης τάξης. Η Skolemized μορφή του w ανακτάται ως εξής: Αντικαθιστάται κάθε υπαρξιακά ποσοδεικτούμενη μεταβλητή y του w από μια συνάρτηση της μορφής $\sigma(x_1,\ldots,x_n)$ όπου τα $(x_1),\ldots,(x_n)$ είναι όλες οι καθολικά ποσοδεικτούμενες μεταβλητές που προηγούνται της (Ey) στο πρόθεμα του w. Το σ πρέπει να είναι διαφορετικό από κάθε άλλο γράμμα του αλφαβήτου A και των γραμμάτων συναρτήσεων που χρησιμοποιούνται ως τώρα. Μετά από αυτήν την αντικατάσταση για όλες τις υπαρξιακά ποσοδεικτούμενες μεταβλητές, αφαιρούνται οι υπαρξιακοί ποσοδείκτες του w. Το αποτέλεσμα είναι ένα τύπος χωρίς ποσοδείκτες. Τα νέα γράμματα συνάρτησης όπως το σ που εισάγονται ονομάζονται *Skolem συναρτήσεις*.

Η *Skolemized μορφή ενός default*:

$$\frac{\alpha(x) : M\beta_1(x),\ldots, M\beta_m(x)}{w(x)}$$

επιτυγχάνεται με την αντικατάσταση του καλά ορισμένου τύπου w(x) από την Skolemized μορφή του (x)w(x).

Παράδειγμα: Η Skolemized μορφή του:

$$\frac{(Ey)(z)P(x,y,z) : M(Ey)Q(x,y)}{(Ey)(z)(Ew)R(x,y,z,w)}$$

θα είναι:

$$\frac{(Ey)(z)P(x,y,z) : M(Ey)Q(x,y)}{R(x,f(x),z,g(x,z))}$$

όπου οι f και g είναι Skolem συναρτήσεις. Θα πρέπει να τονιστεί ότι σε ένα default για την δημιουργία της Skolemized μορφής του μόνο το επακόλουθο μετατρέπεται.

Μια default θεωρία (D, W) είναι σε Skolemized μορφή αν και μόνο αν όλα τα defaults που ανήκουν στο σύνολο D και όλοι οι καλά ορισμένοι τύποι του W είναι σε Skolemized μορφή.

Έστω ότι το G είναι το σύνολο των γραμμάτων των συναρτήσεων. Τότε το H(G) είναι το μικρότερο σύνολο όρων όπου $\alpha \in H(G)$ για κάθε γράμμα σταθεράς $\alpha \in G$, και αν τα $t_1,\ldots,t_n \in H(G)$ και $g \in G$ είναι ένα γράμμα συνάρτησης τάξης n τότε $g(t_1,\ldots,t_n) \in H(G)$. Δηλαδή απλά το σύνολο H(G) περιέχει όλους τους κατασκευάσιμους όρους από τα γράμματα συναρτήσεων του G.

Αν η Δ = (D, W) είναι μια default θεωρία σε Skolemized μορφή και περαιτέρω αν το Σ είναι το σύνολο των Skolem συναρτήσεων της Δ και το F το σύνολο γραμμάτων συναρτήσεων του A τότε ορίζεται το σύνολο όλων των δυνατών οντοτήτων του $H(F \cup \Sigma)$ των defaults κανόνων του D ως εξής:

$CLOSED-DEFAULTS(D) = \{d(g) \mid d(x) \in D$ και g είναι μια πλειάδα από δυνατούς όρους του $H(F \cup \Sigma)\}$.

Extension:

Ορισμός: Το E είναι μια extension της Δ = (D, W) αν και μόνο αν το E είναι μια extension της default θεωρίας κλειστού κόσμου:

CLOSED(Δ) = (CLOSED-DEFAULTS(Δ), W).

Κεφάλαιο 3^{o}
Θεωρία του Αντωνίου.

Εισαγωγή: Εύλογος Συλλογισμός (Default Reasoning).

Όταν ένα έξυπνο, έμπειρο σύστημα προσπαθεί να επιλύσει ένα πρόβλημα, μπορεί να είναι δυνατό να βασιστεί σε πλήρη γνώση για το πρόβλημα και το κύριο έργο του είναι να εξάγει τα σωστά συμπεράσματα χρησιμοποιώντας την κλασική συλλογιστική. Σε αυτές τις περιπτώσεις, η χρήση της κλασικής κατηγορηματικής λογικής είναι επαρκής. Ωστόσο, μπορεί ένα σύστημα να έχει ελλιπείς πληροφορίες είτε γιατί δεν είναι διαθέσιμες είτε γιατί θα πρέπει να ανταποκριθεί γρήγορα και δεν υπάρχει ο χρόνος για τη συλλογή των σχετικών πληροφοριών. Η κλασική λογική πράγματι έχει κάποιο τρόπο αναπαράστασης και συλλογισμού σε ορισμένες περιπτώσεις μη πλήρης γνώσης. Όμως στις περιπτώσεις που πρόσθετη πληροφορία πρέπει να συλλεχθεί για να ξεπεραστούν οι τυχόν ελλείψεις δεδομένων και να παρθούν αποφάσεις είναι αναγκαίο να υπάρχει η δυνατότητα εξαγωγής πιθανών εικασιών, οι οποίες στην πραγματικότητα είναι η εύλογη συλλογιστική που στηρίζεται σε περιεκτικούς, υποθετικούς κανόνες (rules of thumb) που ονομάζονται *defaults*.

Έτσι, οι αποφάσεις που ουσιαστικά βασίζονται σε υποθέσεις μπορεί να αποδειχθούν ότι είναι λάθος κατά την εισαγωγή νέας γνώσης και να πρέπει να ακυρωθούν και το σύστημα να οπισθοχωρήσει σε μία προηγούμενη κατάσταση. Το φαινόμενο αυτό της κατάργησης κάποιων προηγούμενων συμπερασμάτων ονομάζεται *μη μονοτονικότητα*.

Εύλογος Συλλογισμός (Default Reasoning).

Η έννοια ενός default κανόνα:

Η *χρήση* ενός default κρίνεται αναγκαία σε περιπτώσεις όπου:

1. Η εξαγωγή συμπεράσματος βασίζεται σε υποθέσεις.
   Παράδειγμα: Ένας κανόνας που χρησιμοποιείται από τους οργανωτές ποδοσφαίρου στην Γερμανία είναι: 'Ένας ποδοσφαιρικός αγώνας μπορεί να πραγματοποιηθεί εκτός κι αν έχει χιόνι στο στάδιο'. Αυτός ο κανόνας αναπαριστάται με default ως εξής:
   $$\frac{football : \neg snow}{takePlace}$$
   Η μετάφραση αυτού του κανόνα είναι: 'Αν δεν υπάρχει πληροφόρηση ότι θα υπάρχει χιόνι στο γήπεδο, είναι λογικό να υποτεθεί $\neg snow$ και να εξαχθεί το συμπέρασμα ότι ο κανόνας θα γίνει. Όμως αν συνέβη χιονοθύελλα ένα βράδυ πριν την διεξαγωγή του αγώνα η παραπάνω υπόθεση δεν είναι ορθή καθώς υπάρχει ξεκάθαρη πληροφορία ότι υπάρχει χιόνι και δεν είναι σωστή η αποδοχή του $\neg snow$ και έτσι δεν μπορεί να εφαρμοστεί το default. Σε αυτήν την περίπτωση θα πρέπει να γίνει οπισθοχώρηση και να ακυρωθεί το παραπάνω συμπέρασμα (μη μονοτονία).
   Ένας άλλος τρόπος αναπαράστασης του κανόνα είναι με χρήση του default:
   $$\frac{football : takePlace}{takePlace}$$
   μαζί με τον κλασικό κανόνα (εξαίρεση):
   $$snow \rightarrow \neg takePlace$$
2. Γίνεται μοντελοποίηση πρωτότυπου συλλογισμού, το οποίο σημαίνει ότι τα περισσότερα στιγμιότυπα μιας θεωρίας έχουν την ίδια ιδιότητα.
   Παράδειγμα: Η δήλωση ότι: 'Τα περισσότερα παιδιά έχουν γονείς οι οποίοι ζουν' μπορεί να εκφραστεί με το default:

$$\frac{child(X) : hasParents(X)}{hasParents(X)}$$

3. Υπάρχει συλλογισμός για την αποφυγή ρίσκου. Αφορά περιπτώσεις όπου η κατάληξη σε ένα συμπέρασμα απαιτείται ακόμα και όταν δεν είναι το πιο λογικό επειδή μια άλλη απόφαση οδηγεί σε μια καταστροφή.
   Παράδειγμα: Η κύρια αρχή της δικαιοσύνης στους δυτικούς πολιτισμούς είναι: 'Με την απουσία στοιχείων για το αντίθετο θεωρείται ότι ο κατηγορούμενος είναι αθώος'. Αυτό με την χρήση ενός default θα έχει τη μορφή:

$$\frac{accused(X) : innocent(X)}{innocent(X)}$$

Τα defaults μπορούν να χρησιμοποιηθούν για να μοντελοποιήσουν την *Υπόθεση Κλειστού Κόσμου (Closed World Assumption)* του Reiter (Κεφάλαιο 2^o). Σύμφωνα με αυτήν την υπόθεση, μια περιοχή εφαρμογής περιγράφεται με τη βοήθεια κάποιων αξιωμάτων (γεγονότα συσχέτισης, εξισώσεις, κανόνες) με την παρακάτω έννοια: 'ένα ground γεγονός (δηλαδή μη παραμετροποιημένη δήλωση σχετική με αντικείμενα) θεωρείται εσφαλμένο στο πεδίο του προβλήματος αν δεν εξάγεται από τα αξιώματα'. Η αναπαράσταση με την χρήση ενός default για κάθε ground άτομο φ είναι:

$$\frac{true : \neg\varphi}{\neg\varphi}$$

και ουσιαστικά αυτό σημαίνει: 'Αν είναι συνεπές να υποτεθεί $\neg\varphi$ (που είναι ίδιο με το να μην υπάρχει απόδειξη για το φ) τότε συμπεραίνεται ότι $\neg\varphi$.

Το συντακτικό μιας Default Λογικής:

Μια default θεωρία T είναι ένα ζεύγος (W, D) που αποτελείται από ένα σύνολο W τύπων κατηγορηματικής λογικής (ονομάζονται γεγονότα (facts) ή αξιώματα (axioms) της T) και ένα υπολογίσιμο σύνολο D από defaults. Ένα default έχει τη μορφή:

$$\frac{\varphi : \psi_1, \dots, \psi_n}{\chi}$$

όπου τα $\varphi, \psi_1, \dots, \psi_n, \chi$ είναι κλειστή τύποι κατηγορηματικής λογικής και $n \succ 0$. Ο τύπος φ ονομάζεται προαπαιτούμενο (prerequisite) και συμβολίζεται pre(δ), τα $\psi_1, \dots, \psi_n$ ονομάζονται αιτιολογίες (justifications) και συμβολίζονται ως just(δ) και το χ ονομάζεται επακόλουθο (consequent) και συμβολίζεται cons(δ).

Ένα default ονομάζεται κανονικό (normal) όταν έχει τη μορφή:

$$\frac{\varphi : \psi}{\psi}$$

Ένα ανοιχτό, ελεύθερο (open) default ερμηνεύεται ως ένα default schema που σημαίνει ότι αντιπροσωπεύει ένα σύνολο από defaults. Ένα default schema μοιάζει με ένα default, με τη διαφορά ότι τα $\varphi, \psi_1, \dots, \psi_n, \chi$ είναι αυθαίρετοι τύποι κατηγορηματικής λογικής (περιλαμβάνουν ελεύθερες μεταβλητές). Ένα default schema ορίζει ένα σύνολο από defaults της μορφής:

$$\frac{\varphi\sigma : \psi_1\sigma, \dots, \psi_n\sigma}{\chi\sigma}$$

για όλες τις δυνατές αντικαταστάσεις σ οι οποίες αναθέτουν τιμές στις ελεύθερες μεταβλητές που υπάρχουν στο schema. Αυτό σημαίνει ότι οι ελεύθερες μεταβλητές ερμηνεύονται σαν να είναι καθολικά υπολογίσιμες από το όλο default schema.

Παράδειγμα: Έστω το default schema:

$$\frac{bird(X) : flies(X)}{flies(X)}$$

και τα γεγονότα:

$$bird(tweety)$$
$$bird(sam)$$

τότε η default θεωρία αναπαριστάται:

$(\{bird(tweety), bird(sam)\}, \{bird(tweety) : flies(tweety) / flies(tweety), bird(sam) : flies(sam) / flies(sam)$

Defaults και Extensions:

Σημασιολογία ενός Default: Η έννοια ενός default $\varphi : \psi_1,...,\psi_n / \chi$ συνοψίζεται στο εξής: Αν είναι γνωστό το φ και είναι συνεπές να υποθέσουμε τα $\psi_1,...,\psi_n$ τότε μπορούμε να συμπεράνουμε το χ. Ο επίσημος ορισμός του, λοιπόν, θα είναι: Ένα default $\varphi : \psi_1,...,\psi_n / \chi$ είναι εφαρμόσιμο σε ένα κλειστό συμπερασματικά σύνολο από τύπους Ε αν και μόνο αν $\varphi \in$ Ε και $\neg\psi_1 \notin$ Ε,....$\neg\psi_n \notin$ Ε.

Extensions: Οι extensions (επεκτάσεις) αντιπροσωπεύουν πιθανές όψεις του κόσμου, η κατασκευή των οποίων βασίζεται στις δοθείσες default θεωρίες, και προσπαθούν να επεκτείνουν το σύνολο των γνωστών γεγονότων με λογικές εικασίες που εξάγονται από τους διαθέσιμους default κανόνες.

Ιδιότητες ενός extension: Οι επιθυμητές ιδιότητες που πρέπει να έχει ένα extension Ε είναι:

1. Θα πρέπει να περιλαμβάνει το σύνολο W των γεγονότων αφού το W περιλαμβάνει την σίγουρη πληροφορία που είναι διαθέσιμη: $W \subseteq E$.
2. Θα πρέπει να είναι συμπερασματικά κλειστή καθώς δεν είναι επιθυμητή η αποφυγή της κλασικής λογικής συλλογιστικής. Στην πραγματικότητα, στόχος είναι η εξαγωγή κι άλλων συμπερασμάτων και γι' αυτό εφαρμόζονται οι default κανόνες. Επίσημα: E = Th(E), όπου το Th δείχνει το συμπερασματικό κλείσιμο.
3. Θα πρέπει να είναι κλειστή κατά την εφαρμογή των default κανόνων που βρίσκονται στο σύνολο D. Αυτό σημαίνει ότι η διακοπή εφαρμογής των default δεν είναι επιθυμητή εκτός κι αν υπάρξει αναγκαιότητα να γίνει. Η εξήγηση σε αυτό είναι ότι δεν υπάρχει κάποιος λόγος να διακοπεί η εφαρμογή των defaults από τη στιγμή που υπάρχουν κανόνες που μπορούν να εφαρμοστούν. Οι extensions, λοιπόν, είναι μέγιστες πιθανές όψεις κόσμου.

Αυτές οι ιδιότητες δεν είναι όμως επαρκείς για την εύρεση ενός extension καθώς δεν δίνεται καμία πληροφορία για το ποιοι τύποι θα πρέπει να εξαιρεθούν από το extension.Θα πρέπει να σημειωθεί, λοιπόν, ότι κατά τον υπολογισμό ενός extension απαιτείται να βρίσκεται η ελάχιστη που ικανοποιεί αυτές τις ιδιότητες και να ελέγχεται αν κάθε συμπέρασμα που εξάγεται είναι λογικά συνεπές.

Λειτουργικός ορισμός των Extensions:

Για μία δοθείσα default θεωρία T = (W, D) έστω ότι το Π = (δ_0,δ_1,...) είναι μια ακολουθία μετρήσιμη ή μη από defaults που ανήκουν στο D χωρίς πολλαπλές εμφανίσεις. Το *Π* θεωρείται κάτι σαν μια πιθανή σειρά εφαρμογής κάποιων default κανόνων. Συμβολίζουμε το αρχικό τμήμα του Π μεγέθους κ ως Π(κ). Με κάθε τέτοια ακολουθία Π συσχετίζουμε δύο σύνολα από τύπους λογικής πρώτης τάξης, τα In(Π) και Out(Π) όπου:

- Το *In(Π)* είναι το $Th(W \cup \{cons(\delta) \mid \delta$ υπάρχει στο Π}. Δηλαδή το In(Π) συλλέγει τις πληροφορίες που κερδίζονται από την πυροδότηση των default κανόνων στο Π και αναπαριστά την ισχύων βάση γνώσης μετά την εφαρμογή των defaults του Π.
- Το Out(Π) = $\neg\psi \mid \psi \in just(\delta)$ για κάποια defaults που υπάρχουν στο Π}. Ουσιαστικά συλλέγει τύπους που δεν πρέπει να γίνουν αληθής και που δεν πρέπει να γίνουν μέρος της βάσης γνώσης ακόμα και μετά την μετέπειτα εφαρμογή των άλλων defaults.

Παράδειγμα: Έστω ότι υπάρχει η default θεωρία T = (W, D) με:

$$W = \{\alpha\}$$

και με D που περιλαμβάνει τα εξής:

$$\delta_1 = \frac{a : \neg b}{\neg b}, \quad \delta_2 = \frac{b : c}{c}.$$

- Αν το σύνολο Π = (δ_1) τότε:

$$In(\Pi) = Th(\{a, \neg b\}) \text{ και } Out(\Pi) = \{b\}.$$

- Αν το σύνολο Π = (δ_2,δ_1) τότε:

$$In(\Pi) = Th(\{a, c, \neg b\}) \text{ και } Out(\Pi) = \{\neg c, b\}.$$

Στο προηγούμενο παράδειγμα, τα (δ_2,δ_1) δεν μπορούν να εφαρμοστούν με αυτήν την σειρά καθώς το δ_2 δεν μπορεί να εφαρμοστεί αφού το $b \notin In(()) = Th(W) = Th(\{a\}$ που είναι η βάση γνώσης πριν την προσπάθεια ενεργοποίησης του δ_2. Από την άλλη πλευρά, δεν υπάρχει κανένα πρόβλημα με το Π =(δ_1), και σε αυτήν την περίπτωση λέμε ότι το Π είναι μια διαδικασία (process) της T.

Ο επίσημος ορισμός της διεργασίας μιας default θεωρίας T είναι: Η Π ονομάζεται μια *διαδικασία (process)* της T αν και μόνο αν το δ_κ είναι εφαρμόσιμο στο In(Π(κ)), για κάθε κ τέτοιο ώστε το δ_κ να μην υπάρχει ακόμη στη Π.

Επιπλέον, γνωρίζοντας ότι το Π είναι μια διαδικασία της T ορίζουμε τα εξής:

- Η Π είναι *επιτυχής (successful)* αν και μόνο αν $In(\Pi) \cap Out(\Pi) = \{\}$, διαφορετικά είναι αποτυχημένη.
- Η Π είναι *κλειστή (closed)* αν και μόνο αν για κάθε $\delta \in D$ το οποίο είναι εφαρμόσιμο στο In(Π) ήδη συμπεριλαμβάνεται στην Π. Οι κλειστές διαδικασίες ανταποκρίνονται στην επιθυμητή ιδιότητα των extensions να είναι κλειστές κάτω από την εφαρμογή των defaults που περιέχει το D.

Παράδειγμα: Θεωρούμε την εξής default θεωρία T = (W, D) με:

$$W=\{\alpha\}$$

και με D που περιλαμβάνει τα εξής:

$$\delta_1 = \frac{a : \neg b}{\neg b}, \quad \delta_2 = \frac{true : c}{b}.$$

Τότε η:

- Π_1 = (δ_1) είναι επιτυχής αλλά όχι κλειστή αφού το δ_2 μπορεί να εφαρμοστεί στο In(Π_1) = Th({ a, d }).
- Π_2 = (δ_1, δ_2) είναι κλειστή αλλά όχι επιτυχής καθώς και το In(Π_2) = Th({ a, d, b }) και το Out(Π_2) = Th({ $b, \neg c$ }) περιέχουν το b.
- Π_3 = (δ_2) είναι και επιτυχής και κλειστή διαδικασία της T (extension).

1ος Ορισμός της extension: Ένα σύνολο από τύπους E είναι μια extension της default θεωρίας T αν και μόνο αν υπάρχει μια επιτυχής και κλειστή διαδικασία Π της T για την οποία ισχύει E = In(Π). (Antoniou, Sperscchneider 1994).

Έχοντας μια default θεωρία T = (W, D), λέμε ότι η E έχει ένα ημι-επαγωγικό ορισμό

(quasiinductive definition) στην T αν και μόνο αν $E = \cup_i E_i$, όπου $E_o = Th(W)$ και το $E_{i+1} = Th(E_i \cup \{cons(\delta)\}\delta \in D)$ είναι εφαρμόσιμο στο E_i όσον αφορά το σύνολο αντιλήψεων E.
2ος Ορισμός της extension: Η E είναι μια extension της T να και μόνο αν η E έχει ένα ημι – επαγωγικό ορισμό στην T.(Επιρροή από Reiter).

Δέντρο Διαδικασίας (Process Tree):

Είναι συχνά χρήσιμη η αναπαράσταση όλων των πιθανών διαδικασιών με ένα απλοποιημένο τρόπο με την χρήση ενός δέντρου, το οποίο ονομάζεται *δέντρο διεργασίας* της δοθείσας default θεωρίας T. Οι κόμβοι από το δέντρο έχουν δύο ετικέτες που είναι δύο σύνολα τύπων, ένα In-set (στα αριστερά του κόμβου) και ένα Out-set (στα δεξιά του κόμβου). Οι ακμές ανταποκρίνονται στις εφαρμογές των defaults, και έχουν σαν ετικέτα το default κανόνα που εφαρμόζεται. Τα μονοπάτια του δέντρου ξεκινώντας από τη ρίζα αναπαριστούν της διαδικασίες της T.

Μερικά Παραδείγματα:

1ο. Έστω ότι έχουμε την default θεωρία T = (W, D) με W = {} και D = $\{true : a / \neg a\}$. Το δέντρο διαδικασίας θα είναι:

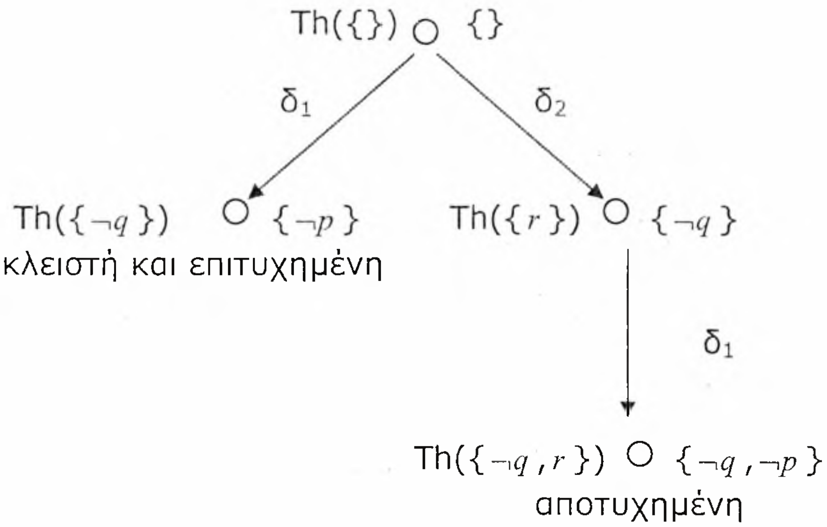
$$
\begin{array}{c}
Th(\{\}) \;\circ\; \{\} \\
\Downarrow \\
Th(\{\neg a\}) \;\circ\; \{\neg a\} \\
\text{αποτυχημένη}
\end{array}
$$

Από το δέντρο είναι κατανοητό ότι η T δεν έχει καμία extension. Πράγματι το default μπορεί να εφαρμοστεί γιατί δεν υπάρχει τίποτα που να εμποδίζει να θεωρηθεί το α. Όμως, μόλις εφαρμοστεί ο κανόνας η άρνηση του α εισάγεται στην βάση γνώσης και έτσι το default από μόνο του ακυρώνει την εφαρμογή του αφού και το In-set και το Out-set περιλαμβάνουν το $\neg a$. Θα πρέπει να τονίσουμε ότι δεν απαιτείται όλες οι default θεωρίες να έχουν extensions.

2ο. Έστω ότι έχουμε την default θεωρία T = (W, D) με W = {} και D = $\{\delta_1,\delta_2\}$ όπου:

$$\delta_1 = \frac{true : p}{\neg q} \text{ και } \delta_2 = \frac{true : q}{r}.$$

Το δέντρο διαδικασίας που φαίνεται παρακάτω δείχνει ότι η T έχει ακριβώς μια extension, την Th($\{\neg q\}$). Το δεξί μέρος του δέντρου δείχνει ένα παράδειγμα όπου η εφαρμογή ενός default καταστρέφει την επιτυχημένη εφαρμογή ενός προηγούμενου default: το δ_1 μπορεί να εφαρμοστεί μετά το δ_2, αλλά τότε το $\neg q$ γίνεται μέρος του In-set, το οποίο ήδη υπήρχε στο Out-set.



Ιδιότητες της default θεωρίας.

Ύπαρξη extensions:

Μια default θεωρία μπορεί να μην έχει extensions. Αν μια θεωρία περιλαμβάνει κανόνες που δεν βγάζουν νόημα (για παράδειγμα: $\frac{true : p}{\neg p}$) τότε η συλλογιστική πρέπει να έχει τη δυνατότητα να μην παρέχει καμία απάντηση. Έγκειται, λοιπόν, στην ευθύνη του χρήστη να παρέχει καλά ορισμένη και ορθή πληροφορία με την μορφή κανόνων και γεγονότων που βγάζουν νόημα.

Επιπλέον, σε μερικές περιπτώσεις, θα ήταν προτιμότερη μια ποιο ανεκτή σε λάθη λογική, έτσι ώστε ακόμα κι αν υπάρχει ελλιπής γνώση να μπορεί να μας οδηγεί σε συμπεράσματα.

Όπως έχει ειπωθεί, η default συλλογιστική είναι μη μονότονη. Για παράδειγμα όταν έχουμε την θεωρία $T = (\{\}, \{true : p / p)$ καταλήγουμε σε μία extension την E = Th({p}), αν όμως είχαμε την $T' = (\{\}, \{true : p / p, true : q / \neg q\})$ δεν θα υπήρχε extension. Έτσι η default συλλογιστική σε πολλές περιπτώσεις παραβιάζει και την ημι-μονοτονικότητα.

Αν η μη ύπαρξη μιας extension θεωρείται πρόβλημα θα πρέπει να περιοριστεί η default συλλογιστική σε μια κλάση default λογικής που να εγγυάται την ύπαρξη τουλάχιστον μίας extension ή να αλλάξει ο ορισμός της extension με τέτοιο τρόπο ώστε να έχουν όλες οι default θεωρίες. Μια κλάση που εγγυάται την ύπαρξη extensions είναι οι κανονικές default θεωρίες (normal default theories) οι οποίες ικανοποιούν και την ιδιότητα την ημι-μονοτονικότητας.

Ωστόσο, οι παραπάνω θεωρίες έχουν περιορισμένη εκφραστικότητα και ιδίως όσον αφορά την αλληλεπίδραση μεταξύ των defaults. Για παράδειγμα, έστω ότι έχουμε τα παρακάτω δεδομένα στη βάση γνώσης:

- Ο Βασίλης έχει αποβληθεί από το λύκειο.
- Τυπικά, όσοι έχουν αποβληθεί από το λύκειο είναι ενήλικες.
- Τυπικά, οι ενήλικες είναι και εργαζόμενοι.

Η αναπαράσταση αυτών των στοιχείων με default λογική θα είναι: $T = (\{dropout(bill)\}, \{dropout(X) : adult(X) / adult(X), adult(X) : employed(X) / employed(X)\})$.

Η T έχει μία extension την:

$$E = Th(\{dropout(bill), adult(bill), employed(bill)\}).$$

Είναι αποδεκτό να δεχτούμε ότι ο Βασίλης είναι ενήλικας, αλλά είναι αντιδιαισθητικό να δεχτούμε ότι είναι και εργαζόμενος. Έτσι, θέλουμε να αποφύγουμε την εφαρμογή του δεύτερου default στην περίπτωση που αναφερόμαστε σε αποβαλλόμενους από το λύκειο. Αυτό επιτυγχάνεται με την αλλαγή του default ως εξής:

$$\frac{adult(X) : employed(X) \wedge \neg dropout(X)}{employed(X)}$$

Αυτός ο κανόνας, όμως δεν είναι κανονικός. Defaults αυτής της μορφής ονομάζονται *ημι-κανονικά*.

Συνέπεια στο συνδυασμό αιτιολογιών (Joint Consistency of Justifications):

Οι αιτιολογίες δεν απαιτείται να σχηματίζουν ένα σύνολο από συνεπή πιστεύω, χρησιμοποιούνται για την εξαγωγή συμπερασμάτων. Στις περισσότερες περιπτώσεις η μη συνέπεια με το συνδυασμό διαφόρων αιτιολογιών δεν δημιουργεί πρόβλημα. Ωστόσο, υπάρχουν θεωρίες που παραπάνω μη συνέπεια οδηγεί σε λάθος συμπεράσματα. Για παράδειγμα, η default θεωρία του Poole η οποία χρησιμοποιεί το default ότι ένα χέρι (a ή b) του ρομπότ είναι χρήσιμο εκτός κι αν είναι σπασμένο, και επιπλέον το γεγονός ότι ένα από τα δυο χέρια είναι σπασμένο. Με αυτήν την πληροφόρηση δεν θα έπρεπε να καταλήξουμε ότι και τα δυο χέρια είναι χρήσιμα.

Η Default λογική αντιμετωπίζει αυτό το παράδειγμα ως εξής. Έστω ότι έχουμε την default θεωρία T = (W,D) με $W = \{broken(a) \vee broken(b)\}$ και με D που περιλαμβάνει τα εξής:

$$\frac{true : usable(a) \wedge \neg broken(a)}{usable(a)},$$

$$\frac{true : usable(b) \wedge \neg broken(b)}{usable(b)}.$$

Αφού δεν έχουμε κάποια έγκυρη πληροφορία ότι το α είναι σπασμένο μπορεί να εφαρμοστεί ο πρώτος κανόνας default και να προκύψει η $E' = Th(W \cup \{usable(a)\})$. Αφού η Ε' δεν περιλαμβάνει το broken(b) μπορεί να εφαρμοστεί και ο δεύτερος κανόνας και να προκύψει $E' = Th(W \cup \{usable(a), usable(b)\})$. Αυτό το αποτέλεσμα είναι ανεπιθύμητο αφού υπάρχει το γεγονός ότι ένα από τα δυο χέρια είναι σπασμένο.

Η συνέπεια στο συνδυασμό αιτιολογιών μειώνει την εκφραστικότητα των default θεωριών: κάτω από αυτήν την ιδιότητα κάθε default με μερικές αιτιολογίες $\varphi : \psi_1, ..., \psi_n / \chi$ είναι ισοδύναμο με το τροποποιημένο default: $\varphi : \psi_1 \wedge ... \wedge \psi_n / \chi$ το οποίο έχει μια αιτιολογία. Ουσιαστικά, στις default λογικές που χρησιμοποιούν την συνέπεια των συνδυασμένων αιτιολογιών είναι αδύνατο να εκφραστούν default κανόνες της μορφής 'Στην περίπτωση που δεν ξέρουμε το p (εννοώντας ότι δεν ξέρουμε ούτε το p ούτε το $\neg p$) καταλήγουμε ότι ισχύει το q.' Η αναπαράσταση του παραπάνω με default θα ήταν:
$true : p, \neg p / q$ και αν απαιτείται συνέπεια στο συνδυασμό των αιτιολογιών τότε ο κανόνας αυτός δεν θα εφαρμοστεί ποτέ.

Αθροιστικότητα και Λήμματα (Cumulativity and Lemmas):

Έστω ότι το D είναι ένα αμετάβλητο, μετρήσιμο σύνολο από defaults. Για ένα τύπο φ και ένα σύνολο από τύπους W ορίζουμε το $W|\!\!-\!\!-_D \varphi$ αν και μόνο αν το φ συμπεριλαμβάνεται σε όλες τις extensions της default θεωρίας (W, D). Η αθροιστικότητα είναι η παρακάτω ιδιότητα:
Αν $W|\!\!-\!\!-_D \varphi$, τότε για όλα τα ψ:

$$W|\!\!-\!\!-_D \psi \Leftrightarrow W \cup \{\varphi\}|\!\!-\!\!-_D \psi.$$

Δηλαδή αν μεταφράσουμε το φ ως λήμμα η αθροιστικότητα υποστηρίζει ότι ο ίδιος τύπος μπορεί να αποκτηθεί από το W όσο και από το $W \cup \{\varphi\}$.

Η default θεωρία δεν σέβεται την αθροιστικότητα. Για παράδειγμα, έστω ότι έχουμε την θεωρία T = (W, D) με W ={} και D αποτελούμενο από τα εξής:

$$\frac{true : a}{a}, \frac{a \vee b : \neg a}{\neg a}.$$

Η μόνη extension της T είναι Th({α}). Προφανώς W|--Dα. Από $a \vee b \in Th(\{a\})$ παίρνουμε ότι W|--D $a \vee b$. Αν είχαμε $W' = \{a \vee b\}$, τότε η default θεωρία (W', D) έχει δύο extensions, Th({α}) και Th({¬α,b}) και έτσι $W \cup \{a \vee b\}$ |--Dα δεν ισχύει.

Ένας επίσημος ορισμός του λήμματος είναι ο εξής:

Έστω ότι η Π_χ είναι μια μη κενή, επιτυχής διαδικασία της T, ελάχιστη με την ιδιότητα $\chi \in In(\Pi_\chi)$. Ένα default λήμμα δ_χ που να ανταποκρίνεται στο χ είναι το default:

$$\frac{true : \psi_1, \ldots, \psi_n}{\chi}$$

όπου $\{\psi_1, \ldots, \psi_n\} = \{\psi \mid \psi \in just(\delta)\}$ για ένα δ που βρίσκεται στην Π_χ. Αυτό το default συλλέγει όλες τις υποθέσεις που χρησιμοποιήθηκαν για να καταλήξουμε στο χ.

Θεώρημα: Έστω ότι το χ συμπεριλαμβάνεται σε μια extension της T και το δ_χ είναι το αντίστοιχο default λήμμα. Τότε κάθε extension της T είναι και μια extension της $T' = (W, D \cup \{\delta_\chi\})$, και αντίστροφα.

Έτσι, είναι δυνατό να αναπαραστήσουμε λήμματα στην Default Λογική, όχι σαν γεγονότα (που απαιτεί η αθροιστικότητα) αλλά ως defaults, που φαίνεται πιο φυσικό, αφού τονίζει την φύση ενός λήμματος που αποδεικνύεται μεν αλλά μόνο παρέχοντας και τη δυνατότητα ακύρωσης (defeasibly).

Κεφάλαιο 4º
Αλγόριθμος του Prakken.

Εισαγωγή.

Έχουν προταθεί δυο τρόποι υλοποίησης της αναιρέσιμης δεοντικής συλλογιστικής που και οι δυο βασίζονται στην άποψη ότι θέματα συγκρουόμενων υποχρεώσεων ή ηθικών διλλημάτων πρέπει να επιλύονται με την βοήθεια της μη μονότονης συλλογιστικής. Η πρώτη θεωρία αναπτύσσει μια ειδική μη μονότονη λογική για δεοντικές δηλώσεις (Horty). Επειδή, όμως, έχει πολλούς περιορισμούς θεωρείται καλύτερη η δεύτερη προσέγγιση που συνδυάζει μια ήδη υπάρχουσα μη μονότονη λογική με μια δεοντική λογική (Δεοντική Λογική: Παράρτημα) (Prakken).

Στην SDL (Standard Deontic Logic) (Παράρτημα), σε ένα τυπικό σύστημα τύπου Kripke-model η εγκυρότητα του D*:

$$\neg(OA \wedge O\neg A)$$

είναι ο μόνος τρόπος για την εγγύηση της εγκυρότητας του D:

$$\neg O \perp .$$

Πολλές προσεγγίσεις δεοντικής λογικής, όμως, δεν αποδέχονται την εγκυρότητα του D* ούτε και του κανόνα:

$$(O(A \vee B) \wedge O \neg A) \rightarrow OB$$

Ο Horty παρατήρησε ότι: Αν ο τελεστής O κάνει έγκυρο τον κανόνα RK:

$$\frac{A_1 \wedge \ldots \wedge A_n \rightarrow B}{OA_1 \wedge \ldots \wedge OA_n \rightarrow OB} \text{ για } n \geq 0$$

αυτός θα είναι έγκυρος μόνο αν είναι έγκυρη και η αρχή (principle) C:

$$(OA \wedge OB) \rightarrow O(A \wedge B).$$

Η μη εγκυρότητα, όμως, του D* κάνει μη έγκυρη και την αρχή C. Έτσι, ο Horty κατέληξε ότι θα θεωρείται έγκυρη η αρχή (μια παραλλαγή της) C μόνο όταν δεν παραβιάζει το D.

Για να επιτύχει το παραπάνω, ο Horty υιοθέτησε την εξής ιδέα: Για να μην χρησιμοποιήσει μια τροπική λογική ασθενέστερη της SDL αντιμετωπίζει τους δεοντικούς κανόνες ως αναιρέσιμους που υπόκεινται σε μη προβλεπόμενες επεκτάσεις και πιθανές συγκρούσεις. Με αυτόν τον τρόπο κατάφερε να κάνει χρήση μη μονότονων λογικών και να καταλήγει σε συμπεράσματα που μπορεί η εισαγωγή νέας γνώσης να τα ακυρώσει.

Μη μονότονη δεοντική λογική του Horty.

Η Λογική:

Χρησιμοποιεί τη θεωρία για default logic του Reiter για να αναπτύξει ο Horty την δική του μη μονότονη δεοντική λογική. Δηλαδή, βασίζεται σε ένα σύνολο F τύπων πρώτης τάξης λογικής και ένα σύνολο Δ default κανόνων της μορφής

A : B / C όπου A : προαπαιτούμενο
B : ισχυρισμός
C : επακόλουθο

Νέα beliefs μπορούν να δημιουργηθούν με την χρήση των default κανόνων με την προϋπόθεση, όμως, να μην προκαλούν ασυνέπεια.

Ορισμός Extension: Σε ένα σύνολο από defaults κανόνες που ανήκουν στο Δ και οι οποίοι έχουν εφαρμοστεί για την εξαγωγή συμπερασμάτων όταν οποιαδήποτε επιπλέον εφαρμογή κάποιου default κανόνα οδηγεί σε μη συνεπή κατάσταση λέμε ότι το σύνολο κανόνων αυτό αποτελεί μια επέκταση της λογικής (F, Δ).

Επιπλέον, στο σύστημα του ο Horty χρησιμοποιεί ένα ought context το (F, Δ_Γ) όπου το F: είναι το σύνολο κανόνων λογικής πρώτης τάξης και αντιπροσωπεύουν την πληροφορία που αφορά γεγονότα και το Γ: είναι ένα σύνολο υπό συνθήκες ought προτάσεων της μορφής O(A/B) και δηλώνει ότι το A είναι υποχρεωτικό στην περίπτωση του B. Θα πρέπει να τονίσουμε ότι το O(A/B) μπορεί να διαβαστεί σαν τα default του Reiter με τη μορφή: B : A / A.

Στην προσπάθεια του ο Horty να εκφράσει ότι πιο ειδικοί κανόνες υπερισχύουν πιο γενικούς ορίζει ότι: ένας υπό συνθήκες κανόνας ought O(A/B) υπερκαλύπτεται στο περιβάλλον (F,Δ_Γ) από ένα κανόνα O(A′/B′) όταν:

- $F \vdash B$, $B' \vdash B$ και $B \neq B'$
- {F,A′,A} είναι μη συνεπές
- {F, A′} είναι συνεπές

Με αυτόν τον τρόπο, μια *υπό συνθήκη extension* του (F,Δ_Γ) ορίζεται ως ένα σύνολο E τέτοιο ώστε όταν υπάρχει ένα άλλο σύνολο S που ορίζεται:

$$S = \{A \mid O(A/B) \in \Gamma,\ F \vdash B,\ O(A/B) \text{ δεν υπερκαλύπτεται στο } (F,\Delta_\Gamma),\ \neg A \notin E\}$$

τότε:

$$E=Th(S \cup \{F\}).$$

Στη συνέχεια, ο Horty αφού απέδειξε ότι κάθε ought context έχει μια υπό συνθήκη extension όρισε τα εξής:

- ο κανόνας O(A/B) γίνεται αληθής με συνέπεια στο Γ αν και μόνο αν A ϵ E για κάποια υπό συνθήκη επέκταση E του (B, Δ_Γ)
- και τον τύπο OA ως O(A / T).

Στην θεωρία λογικής που αναπτύχθηκε παραπάνω, το D είναι έγκυρο σύμφωνα με την τελευταία συνθήκη του S, η περιορισμένη μορφή της αρχής C είναι έγκυρη σύμφωνα με την λογική ότι αν δύο κανόνες ανήκουν σε μία επέκταση τότε και η ένωση τους ανήκει στην επέκταση και το D* είναι μη έγκυρο αφού μπορεί να υπάρξουν πολλές αμοιβαία αποκλειόμενες επεκτάσεις.

Κριτική:

Η προσέγγιση του Horty δεν μπορεί να υποστηρίξει πολλές μορφές δεοντικού συλλογισμού καθώς δεν υπάρχει ικανοποιητική ανάλυση για την έκφραση άμεσων αδειών (permissions). Υποστηρίζει μόνο άδειες με την μορφή συμπερασμάτων (conclusions) που είναι ασθενείς άδειες. Ένα επιπλέον πρόβλημα είναι ότι με τη λογική του Horty μπορούν να δηλωθούν συσχετίσεις μόνο μεταξύ των υποχρεωτικών (ought) δηλώσεων καθώς δεν υπάρχει διαχωρισμός μεταξύ του ποια είναι η περίπτωση και του ποια θα πρέπει να είναι. Ακόμη, δεν μπορεί να υποστηρίξει τον χωρισμό ή την απόσπαση των γεγονότων. Δηλαδή από τον κανόνα O(A/B) και το γεγονός B δεν μπορεί να συμπεράνει την OA. Ούτε και η παραβίαση μιας υποχρέωσης υπάρχει κάποιος τρόπος να δηλωθεί. Τέλος, πρόβλημα αποτελεί και το γεγονός ότι στις δεοντικές λογικές αναιρέσιμοι δεν θεωρούνται μόνο οι δεοντικοί κανόνες μπορούν και άλλοι κανόνες που έχουν δημιουργηθεί από την εφαρμογή άλλων κανόνων να πρέπει να αντιμετωπιστούν ως αναιρέσιμοι. Κάτι που δεν μπορεί να εκφραστεί με τη θεωρία του Horty.

Η λύση των παραπάνω προβλημάτων θα ήταν να χρησιμοποιηθεί ο default συλλογισμός ως έχει και η γλώσσα του (λογική πρώτης τάξης) να επεκταθεί με δεοντικούς τελεστές. Με αυτόν τον τρόπο θα επιτυγχάνονταν διαχωρισμός μεταξύ των default και των δεοντικών υποθέσεων, δεν θα απαιτούνταν η ύπαρξη κάποιας ειδικής δεοντικής μορφής αναιρεσιμότητας και το D* θα γινόταν έγκυρο.

Ηθικά Διλλήματα και Συνέπεια.

Σε αυτό το σημείο θα πρέπει να τονιστεί ότι το D* δεν θα πρέπει να θεωρείται από μια λογική έγκυρο άδικα. Η συνέπειας της περιγραφής μιας κατάστασης δεν θα πρέπει να συγχέεται με την συνέπεια των περιγραφόμενων κανόνων ούτε και να επιτρέπεται η δυνατότητα ενεργοποίησης κανόνων με αντιφατικά δεδομένα (συγκρουόμενοι) που εκφράζουν συνήθως τα ηθικά διλλήματα.

Ωστόσο, σε μερικές περιπτώσεις η εφαρμογή ενός κανόνα που επιφέρει ασυνέπεια σε μια υπάρχουσα κατάσταση δεν θα πρέπει να καταργεί την δυνατότητα εφαρμογής αυτού του κανόνα σε μια άλλη κατάσταση (binding force of rules). Επιπλέον, με τη χρήση της μη μονότονης συλλογιστικής μπορεί να επιτευχθεί εκμετάλλευση της πληροφορίας που παρέχεται από τις συγκρούσεις. Σε πολλές λογικές που έχουν αναπτυχθεί η ύπαρξη αντιθέσεων δεν κάνει το σώμα της πληροφορίας τελείως άχρηστο, σε αυτές μπορεί να γίνει επίλυση των συγκρούσεων χρησιμοποιώντας προτιμήσεις (preferences).

Ένα πλαίσιο εργασίας με επιχειρήματα στην δεοντική default λογική.

Η δεύτερη προσέγγιση για την υλοποίηση της αναιρέσιμης δεοντικής λογικής προτείνει τον συνδυασμό της υπάρχουσας δεοντικής λογικής SDL με μια μη μονοτονική λογική που βασίζεται στο default συλλογισμό και τον επεκτείνει με τα εξής δύο στοιχεία:

- ☐ τον ορισμό της έννοιας των επιχειρημάτων και χρήση τους και
- ☐ την έκφραση προτιμήσεων όταν μια σύγκρουση δημιουργείται.

Τα πλεονεκτήματα αυτής της θεωρίας είναι το ότι η χρήση του default λογισμού κάνει εφικτή την διαφανή σύγκριση της προσέγγισης αυτής με αυτή του Horty και το ότι η ύπαρξη επιχειρημάτων δίνει την δυνατότητα έκφρασης των ηθικών διλημμάτων.

Τροπική Default Λογική (Modal Default Logic):

Δεν υπάρχει κάποιος λόγος που να εμποδίζει την χρήση της default λογικής συνδυαζόμενη με κάποια άλλη μορφή λογικής. Ωστόσο, οι ανοιχτοί default κανόνες του Reiter (περιέχουν ελεύθερες μεταβλητές) χρησιμοποιούν την διαδικασία Skolemization λογικής πρώτης τάξης και αυτή είναι προβληματική στις τροπικές κατηγορηματικές λογικές που δεν εγγυώνται την εγκυρότητα του Barcan τύπου:

$$\Diamond \exists x Px \rightarrow \exists x \Diamond Px$$

Αυτό συμβαίνει γιατί αν πάρουμε την skolemized μορφή του $\Diamond \exists x\, Px$ ως Pa όπου το a είναι μια σταθερά Skolem τότε συμπεραίνουμε ότι $\exists x \Diamond Px$. Έτσι, η μόνη απαίτηση που υπάρχει είναι ότι η προσέγγιση τροπικής λογικής που επεκτείνει την default λογική θα πρέπει να καθιστά έγκυρο τον τύπο Barcan.

Πλαίσιο Εργασίας:

Στο πλαίσιο εργασίας αυτό χρησιμοποιούνται οι κανονικοί κανόνες της μορφής:

A : B / B όπου A : προαπαιτούμενο
B : ισχυρισμός
B : επακόλουθο

με την μορφή των αναιρέσιμων κανόνων:

$$A \Rightarrow B$$

Αποτελείται από τέσσερα βασικά μέρη: την έννοια των επιχειρημάτων, την έννοια της σύγκρουσης ενός επιχειρήματος, την έννοια της σύγκρισης δύο επιχειρημάτων και την έννοια των δικαιολογημένων επιχειρημάτων.

Τα *επιχειρήματα* είναι της ίδιας λογικής με τις default αποδείξεις του Reiter:Έστω D_i το σύνολο των defaults και PRE(D) και CONS(D) το σύνολο των προαπαιτούμενων και των επακόλουθων αντίστοιχα. Τότε ένα επιχείρημα ορίζεται ως μια μη άπειρη ακολουθία $D_0,\ldots\ldots,D_n$ πραγματικών οντοτήτων του D για τα οποία ισχύει:

- Για $1 \leq i \leq n$, F U CONS(D_i) $\vdash$ φ για κάθε φ ϵ PRE(D_{i-1})
- $D_n = \emptyset$
- $\cup_{i=0}^{n}$ CONS(D_i) U F είναι συνεπές

Ουσιαστικά, το D_1 συλλέγει όλα τα defaults που χρησιμοποιούνται άμεσα για την εξαγωγή συμπερασμάτων, το D_2 συλλέγει όλα τα defaults που χρησιμοποιούνται για την παραγωγή των προαπαιτούμενων των defaults του D_1 και αυτό συνεχίζεται μέχρι να καταλήξουμε ότι η δήλωση μας υπάρχει στα γεγονότα.

Για κάθε επιχείρημα A= $D_0,\ldots\ldots,D_n$ ένα υπό επιχείρημα του Α θα είναι ένα επιχείρημα A′= $D_0',\ldots,D_n'$ για το οποίο ισχύει για κάθε i ($1 \leq i \leq n$) $D_i' \subseteq D_i$. Επιπλέον, το A′ είναι ένα ορθό υπό επιχείρημα του Α αν και μόνο αν A′ ≠ A. Κάθε τύπος που υπονοείται από $\cup_{i=0}^{n}$ CONS(D_i) U F είναι ένα επακόλουθο του Α και ένα επακόλουθο είναι τελικό αν και μόνο αν δεν είναι επακόλουθο από ένα υπό επιχείρημα του Α.

Ένα επιχείρημα A_1 *συγκρούεται* με ένα επιχείρημα A_2 αν και μόνο αν

- τα A_1 και A_2 έχουν συγκρουόμενα τελικά συμπεράσματα και
- το A_1 (ή το A_2) δεν έχει κάποιο επακόλουθο φ για το οποίο το ¬φ είναι ένα επακόλουθο από κάποιο υπό επιχείρημα του A_2 (ή του A_1).

Η *σύγκριση* επιτυγχάνεται σύμφωνα με κάποια κριτήρια ή μη καθορισμένα στάνταρ ήττας που καθορίζονται από τον χρήστη του πλαισίου εργασίας και αφορούν κάποιο ελάχιστο σύνολο ιδιοτήτων. (Είσοδος δεδομένων από τον χρήστη).

Το σύνολο των δικαιολογημένων επιχειρημάτων είναι το μικρότερο σύνολο JA επιχειρημάτων για το οποίο ισχύει Α ϵ JA αν και μόνο αν:

- όλα τα ορθά συμπεράσματα του Α ανήκουν στο JA και
- το Α νικά όλα τα επιχειρήματα A′ που "επιτίθενται" του Α και επιπλέον ούτε το A′ ούτε κάποιο υπό επιχείρημά του νικιούνται από κάποιο άλλο επιχείρημα του JA.

Ο ορισμός των δικαιολογημένων επιχειρημάτων χωρίζει τα επιχειρήματα σε τρεις κατηγορίες:

- δικαιολογημένα επιχειρήματα
- επιχειρήματα που ανατρέπονται
- defensible επιχειρήματα: ούτε ανατρέπονται ούτε είναι δικαιολογημένα.

Με το παρακάτω παράδειγμα θα γίνουν κατανοητοί οι τύποι των επιχειρημάτων. Έστω ότι έχουμε ένα γραφειοκρατικό οργανισμό στον οποίο ένας υπάλληλος λέει ότι οι αιτήσεις των πελατών δεν χρειάζεται να απαντιούνται, ενώ ένας ανώτερος υπάλληλος λέει ότι οι γραπτές αιτήσεις θα πρέπει να απαντιούνται. Επιπρόσθετα, υπάρχουν δύο συγκρουόμενα προηγούμενα πιστεύω στο τι υπολογίζεται ως γραπτή αίτηση. Αυτό το παράδειγμα τυποποιείται ως εξής: Έστω ότι έχουμε τη θεωρία Τ = (F, Δ) με F = {f} και Δ = {d_1-d_4} όπου:

d_1 : By_fax ⇒ Written
d_2 : ¬ By_mail ⇒ ¬ Written
d_3 : Request ⇒ ¬ OAnswer
d_4 : Written ⇒ OAnswer και

f : Request ∧ By_fax ∧ (By_fax → ¬ By_mail)

Υποθέτουμε ότι η σχέση ήττας μεταξύ των επιχειρημάτων είναι: Καμία σχέση ήττας δεν υπάρχει μεταξύ των A_1={d_1},∅ και A_2={d_2},∅, ενώ υπολογίζοντας τις βαθμίδες κατάταξης των υπαλλήλων, το επιχείρημα A_4={d_4},{d_1},∅ νικά το αντικρουόμενο επιχείρημα A_3={d_3},∅. Επιπλέον, θα πρέπει να πούμε ότι ορθό υπό επιχείρημα του A_4 είναι το A_1 καθώς και το ∅ ενώ των A_1, A_2, A_3 είναι μόνο το Ø. Παρόλα αυτά, αφού τα A_1 και A_2 δεν νικά κάποιο το άλλο και δεν δηλώνονται ξανά από κάποιο άλλο επιχείρημα, είναι και τα δυο μη δικαιολογημένα. Αυτό κάνει και το A_4 μη δικαιολογημένο. Από την άλλη πλευρά, αφού κανένα επιχείρημα δεν συγκρούεται με κάποιο δικαιολογημένο όλα τα επιχειρήματα είναι defensible.

Αν υποθέσουμε, όμως, ότι κάποια εξουσία αποφασίζει ότι το A_1 νικά το A_2, τότε το A_1 είναι δικαιολογημένο, το οποίο με τη σειρά του κάνει δικαιολογημένο το A_4 και καθώς αυτό το επιχείρημα νικά το A_3 γίνονται όλα τα υπό επιχειρήματά του δικαιολογημένα. Έτσι, τα A_2 και A_3 ανατρέπονται.

Σύγκριση των δύο προσεγγίσεων:

Η δεύτερη προσέγγιση λύνει πολλά από τα προβλήματα της προσέγγισης του Horty:

1. Η defensible συλλογιστική δεν επικυρώνει το D* ακόμα και αν το καθιστά έγκυρο η δεοντική λογική, επικυρώνει, όμως, την περιορισμένη αρχή C.
2. Το πλαίσιο εργασίας μπορεί να εκφράσει ικανοποιητικά το διαχωρισμό γεγονότων.
3. Τα δεοντικά defeasible επιχειρήματα δεν συγχέονται με αυτά που αφορούν γεγονότα και έτσι επιλύονται δύο προβλήματα του Horty: Μπορούν να εκφραστούν άμεσες άδειες καθώς και συγκρούσεις μεταξύ αδειών και υποχρεώσεων και δεν δημιουργούνται προβλήματα με την αλυσιδωτή χρήση γεγονότων και τα δεοντικά defaults.
4. Η αναφορά στο ειδικότερο δεν είναι το μοναδικό κριτήριο για την σύγκριση επιχειρημάτων όπως στου Horty.
5. Αν και είναι αδύνατο στο πλαίσιο εργασίας από δύο defaults να δημιουργηθεί ένα νέο default (όπως και στον Horty), ωστόσο, μπορεί αν υπάρχουν οι προϋποθέσεις στο F να παραχθεί ένα επιχείρημα με τα δύο αυτά defaults

Ωστόσο, αυτό που δεν επιτυγχάνεται είναι η εγκυρότητα του D* που ήταν ένας από τους μελλοντικούς στόχους του Horty.

Συμπέρασμα:

Η γενική προσέγγιση τυποποίησης του defeasible δεοντικού λογισμού δεν είναι μόνο φιλοσοφικά αλλά και μεθοδικά επαρκής καθώς προβλήματα που ανήκουν στο ένα από τα δύο πεδία μπορούν να μελετηθούν απομονωμένα αλλά και οι λύσεις τους μπορούν να βρεθούν εύκολα με τον συνδυασμό τους.

Κεφάλαιο 5º
Αλγόριθμος που υλοποίησα.

Εισαγωγή.

Η βασική ιδέα που στηρίχτηκε ο αλγόριθμος αυτός είναι η επέκταση και βελτιστοποίηση του αλγορίθμου του Αντωνίου για την εύρεση extension και των διαδικασιών Π μιας θεωρίας Τ = (W, D). Με αυτόν τον τρόπο θα προσπαθήσουμε να ξεπεράσουμε κάποιες αδυναμίες του και τον περιορισμό ότι μόνο οι κανονικές θεωρίες default έχουν μια extension σίγουρα.

Χρησιμοποιούμε τον τρόπο εύρεσης των διαδικασιών του Αντωνίου σε συνδυασμό με τη Δικαιολογημένη Default Λογική. Δηλαδή, θεωρούμε ότι μια διαδικασία Π για να αποτελεί το In-Set της μια extension θα πρέπει να είναι επιτυχημένη αλλά όχι απαραίτητα κλειστή. Ακόμα και επιτυχημένες διαδικασίες που η εφαρμογή όλων των άλλων αχρησιμοποίητων default οδηγεί σε ασυνεπή κατάσταση θα θεωρείται μια extension της Τ. Με αυτόν τον τρόπο πολλές λογικές default που κατά τον Αντωνίου δεν έχουν extension τώρα μπορεί να έχουν.

Επιπλέον, ο συνδυασμός του αλγορίθμου του Αντωνίου με την συλλογιστική του Prakken έδωσε νέες δυνατότητες αφού επιχειρείται η συνεύρεση της default συλλογιστικής και των αναιρέσιμων (defensible) κανόνων. Με αυτήν την προσέγγιση κάθε διαδικασία μιας θεωρίας Τ ισοδυναμεί με ένα επιχείρημα όπως ορίστηκαν από τον Prakken που μπορεί να είναι είτε δικαιολογημένο όταν υπερισχύει σε άλλα, είτε καταργήσιμο όταν νικιέται, είτε defensible όταν ούτε είναι δικαιολογημένο ούτε υπό κατάργηση. Ένα σημαντικό πλεονέκτημα είναι ότι οι κανόνες που απαρτίζουν ένα επιχείρημα δεν περιορίζονται σε αυτούς που έχουν κανονική μορφή όπως στο πλαίσιο εργασίας του Prakken, αλλά οι default κανόνες έχουν την μορφή:

A : B / C όπου A : προαπαιτούμενο
B : ισχυρισμός
C : επακόλουθο

Επιπλέον, για την υπερίσχυση ενός κανόνα πάνω σε έναν άλλον χρησιμοποιήθηκαν εκτός από αυτό που προβλέπει ο Prakken και οι προτεραιότητες ως έκφραση προτίμησης επιλογής κάποιου κανόνα έναντι ενός άλλου που είναι και αυτός εφαρμόσιμος.

Δικαιολογημένη Default Λογική (Justified Default Logic).

Η ουσία της προσέγγισης αυτής είναι: Αν έχουμε μια επιτυχής αλλά όχι ακόμη κλειστή διαδικασία, όπως την ορίζει ο Αντωνίου (Κεφάλαιο 3º) και όλοι οι τρόποι επέκτασης της εφαρμόζοντας έναν νέο default κανόνα οδηγούν σε αποτυχία, τότε σταματάμε και δεχόμαστε το τρέχων In-set ως extension. Με άλλα λόγια παίρνουμε πίσω το τελευταίο βήμα που οδηγεί σε αποτυχία.

Παράδειγμα: Έστω ότι έχουμε την default θεωρία Τ = (W, D) με W={holidays, sunday}που σημαίνει ότι γνωρίζουμε ότι είμαστε σε διακοπές και είναι και Κυριακή και με D που περιλαμβάνει τα εξής:

$$\delta_1 = \frac{sunday : goFishing \wedge \neg wakeUpLate}{goFishing}$$

(Αν είναι Κυριακή και μπορώ να πάω για ψάρεμα και δεν έχω ξυπνήσει αργά, τότε πάω για ψάρεμα)

$$\delta_2 = \frac{holidays : wakeUpLate}{wakeUpLate}.$$

(Αν είμαι σε διακοπές και μπορώ να ξυπνήσω αργά, ξυπνάω αργά)

Είναι εύκολο να διαπιστωθεί ότι η Τ έχει ακριβώς μία extension την Th({holidays, sunday,wakeUpLate}). Όμως αν εφαρμόσουμε πρώτα τον δ_1 τότε ο δ_2 μπορεί να εφαρμοστεί και να οδηγήσει σε αποτυχημένη διαδικασία. Με αυτόν τον τρόπο χάνουμε την ενδιάμεση πληροφορία Th({holidays, sunday, goFishing}). Από την άλλη πλευρά, με την Δικαιολογημένη Default Λογική θα σταματούσαμε μετά την εφαρμογή του δ_1, αντί να εφαρμόσουμε και τον δ_2 και να οδηγηθούμε σε αποτυχία και θα δεχόμασταν την Th({holidays, sunday, goFishing}) σαν μια πρόσθετη (τροποποιημένη) extension.

Τεχνικά, αυτό κατορθώνεται με το να δίνουμε έμφαση στις μέγιστες επιτυχημένες διαδικασίες. Έστω ότι η T είναι μια default θεωρία και έστω ότι οι Π και Γ είναι διαδικασίες αυτής. Ορίζουμε $\Pi \prec \Gamma$ αν και μόνο αν το σύνολο των defaults που συμβαίνουν στο Π είναι ένα κανονικό υποσύνολο των defaults που συμβαίνουν στο Γ. Η Π καλείται μια μέγιστη διαδικασία της T αν και μόνο αν είναι επιτυχημένη και δεν υπάρχει άλλη επιτυχημένη διαδικασία Γ για την οποία ισχύει $\Pi \prec \Gamma$.

Ένα σετ από τύπους ονομάζεται μια *τροποποιημένη extension* της T αν και μόνο αν υπάρχει μια μέγιστη διαδικασία Π της T τέτοια ώστε E=In(Π).

Στο προηγούμενο παράδειγμα, η Π = $\{\delta_1\}$ είναι μια μέγιστη διαδικασία: η μοναδική διαδικασία που αυστηρά περιλαμβάνει την Π είναι η Γ = (δ_1, δ_2) που οδηγεί σε αποτυχία. Παρόλα αυτά η Th({holidays, sunday, goFishing}) είναι μια τροποποιημένη extension της T. Η T έχει ακόμα μία τροποποιημένη extension, που είναι και η μόνη extension την Th({holidays, sunday, wakeUpLate}) της T. Προφανώς, κάθε κλειστή και επιτυχημένη διαδικασία είναι μια μέγιστη διαδικασία (αφού δεν μπορούν να εφαρμοστούν νέα defaults).

Θεώρημα: Κάθε extension της default λογικής T είναι μια τροποποιημένη extension της T.

Στο δέντρο διεργασίας μιας default θεωρίας T οι μέγιστες διαδικασίες ανταποκρίνονται είτε σε κλειστούς και επιτυχημένους κόμβους, είτε σε κόμβους n όπου όλα τα άμεσα παιδιά του n οδηγούν σε αποτυχία.

Στο παράδειγμα που ακολουθεί μελετάται η περίπτωση μιας default θεωρίας που δεν έχει extension. Έστω ότι έχουμε την T = (W, D) με W = {} και D = $\{true{:}p/\neg p\}$.Θα πρέπει να τονίσουμε ότι η άδεια διαδικασία είναι μέγιστη (αν και όχι κλειστή), επειδή η εφαρμογή των περίεργων defaults θα οδηγούσε σε αποτυχημένη διαδικασία. Έτσι η Th({}) είναι μια τροποποιημένη extension της T.

Θεώρημα: Κάθε default θεωρία έχει τουλάχιστον μία τροποποιημένη extension. Επιπλέον, η Δικαιολογημένη Default Λογική ικανοποιεί την ημι-μονοτονικότητα.

Προτεραιότητες.

Συχνά είναι επιθυμητό να υπάρχουν προτεραιότητες μεταξύ των default κανόνων που εκφράζουν μια προτίμηση στην εφαρμογή κάποιου κανόνα όταν υπάρχουν πολλοί κανόνες που μπορούν να εφαρμοστούν. Ένας τρόπος είναι να χρησιμοποιηθούν έμμεσα κριτήρια προτίμησης όπως είναι η ιδιαιτερότητα μεταξύ των defaults. Αυτά τα κριτήρια όμως έχουν το πρόβλημα ότι μπορεί να μην είναι επαρκή για να εφαρμοστούν σε πολλούς τομείς εφαρμογής. Για παράδειγμα η ιδιαιτερότητα ενός κανόνα σε σχέση με έναν άλλο δεν μπορεί να εκφράσει το γεγονός ότι οι πιο πρόσφατοι κανόνες προηγούνται των πιο παλιών. Έτσι άλλα είδη προσεγγίσεων θα πρέπει να χρησιμοποιηθούν, στα οποία η πληροφορία προτίμησης να δηλώνεται άμεσα. Δυο τέτοιες προσεγγίσεις είναι οι στατικές και οι δυναμικές προτεραιότητες.

Στατικές Προτεραιότητες:

Στην Default Λογική με Προτεραιότητες ο χρήστης δίνει την σειρά των προτεραιοτήτων με την οποία τα defaults θα πρέπει να εφαρμοστούν στην περίπτωση που πάνω από ένα defaults μπορούν να ενεργοποιηθούν. Στην πιο απλή μορφή, μια αυστηρή καλά ορισμένη σειρά (δηλαδή τα defaults είναι διατεταγμένα κατά φθίνουσα σειρά) χρησιμοποιείται.

Ας θεωρήσουμε ότι χρησιμοποιείται η default θεωρία T = (W, D) με W = {bird, penguin} και με D που περιλαμβάνει τα εξής defaults:

$$\delta_1 = \frac{penguin : \neg flies}{\neg flies}, \ \delta_2 = \frac{bird : flies}{flies}.$$

(δ_1:αν είναι πιγκουίνος και μπορώ να υποθέσω ότι δεν πετάει, τότε δεν πετάει)

(δ_2:αν είναι πουλί και μπορώ να υποθέσω ότι πετάει, τότε πετάει)

Αν υποθέσουμε ότι δίνεται μια ολική σειρά << σύμφωνα με την οποία το δ_1 προτιμάται από το δ_2, δηλαδή $\delta_1 << \delta_2$. Τότε η T σε συνδυασμό με την << θα πρέπει να δεχτεί μόνο την extension Th({bird,penguin, ¬ flies}).

Ορισμός: Αν T = (W, D) είναι μια κανονική default θεωρία και << μια καλά ορισμένη ολική διάταξη του D τότε μια διαδικασία Π = (δ_0, δ_1,...) της T δημιουργείται από την << αν και μόνο αν για όλα τα i, δ_i είναι το << -μικρότερο default από Π- Π[i] το οποίο είναι εφαρμόσιμο στο In(Π[i]), με την προϋπόθεση ότι αυτό το δ_i υπάρχει. Προφανώς η Π είναι κλειστή και είναι και επιτυχημένη αφού η T είναι μια κανονική default θεωρία. Επιπλέον, λέμε ότι η E δημιουργείται από την << αν και μόνο αν E = In(Π) για μια διαδικασία Π που δημιουργείται από την <<.

Στο προηγούμενο παράδειγμα, η << ορίζεται ως $\delta_1 << \delta_2$. Η (δ_1) είναι η διαδικασία που δημιουργείται από την <<, και η Th({bird,penguin, ¬ flies}) είναι η extension που οφείλεται στην <<.

Ασφαλώς, δεν θα έχουμε πάντα ολική διάταξη των defaults. Συχνά τα defaults πρέπει να μένουν μη συγκρίσιμα το ένα με το άλλο. Με άλλα λόγια, επιδιώκοντας να έχουμε σε ολική διάταξη τα defaults αυτό μερικές φορές θα τα κάνει υπερ-καθορισμένα.

Ένα απλό παράδειγμα από την πολιτική. Σύμφωνα με τους συντηρητικούς πολιτικούς, οι φόροι θα πρέπει να κόβονται χωρίς να κόβονται τα έξοδα, το τελευταίο είναι μη απαραίτητο γιατί η μείωση των φόρων υποτίθεται πως οδηγεί στην οικονομική ανάπτυξη. Ακραία συντηρητικοί δηλώνουν τη μείωση των φόρων και των εξόδων επειδή η κυβέρνηση θα πρέπει να είναι όσο πιο επιεικής γίνεται. Σύμφωνα με τους τυπικούς δημοκράτες, ούτε οι φόροι ούτε τα έξοδα θα πρέπει να μειώνονται επειδή πιστεύουν ότι η κυβέρνηση θα πρέπει να παρέχει προγράμματα κοινής πρόνοιας και να δημιουργεί πρόσθετη ζήτηση. Αυτή η πληροφορία μπορεί να εκφραστεί με τα παρακάτω defaults:

$$\delta_1 = \frac{conservative : taxCut \wedge \neg spendingCut}{taxCut \wedge \neg spendingCut},$$

$$\delta_2 = \frac{conservative \wedge radical : taxCut \wedge spendingCut}{taxCut \wedge spendingCut},$$

$$\delta_3 = \frac{socialDemocrat : \neg taxCut \wedge \neg spendingCut}{\neg taxCut \wedge \neg spendingCut}.$$

Διαισθητικά και μόνο είναι σαφές ότι στο δ_2 θα πρέπει να δοθεί μεγαλύτερη προτεραιότητα από το δ_1 γιατί η πληροφορία ότι κάποιος είναι ακραίος συντηρητικός είναι πιο συγκεκριμένη από αυτήν ότι είναι συντηρητικός. Αλλά δεν υπάρχει κάποιος λόγους να ορίσουμε προτεραιότητες ανάμεσα στους δ_1 και δ_3 ή δ_2 και δ_3. Έτσι έχουμε μόνο ότι $\delta_2 < \delta_1$. Τεχνικά μιλώντας, σ' αυτήν την περίπτωση η πληροφορία προτεραιότητας δίνεται με αυστηρή μερική διάταξη < στο σύνολο των defaults.

Ορισμός: T = (W, D,<) είναι μια default λογική με προτεραιότητες αν και μόνο αν η (W, D) είναι μια κανονική default θεωρία και < μια αυστηρή μερική διάταξη στο D. Η E είναι μια PDL - extension της T αν και μόνο αν υπάρχει μια αυστηρή καλά ορισμένη διάταξη << στο D η οποία περιλαμβάνει τη < και δημιουργεί την E.

Στο προηγούμενο παράδειγμα, αν είχαμε W = {conservative, radical} και την < που ορίζεται από $\delta_2 < \delta_1$. Υπάρχουν τρεις καλά ορισμένες διατάξεις στο D που περιλαμβάνουν την <, οι οποίες είναι:

$$\delta_3 << \delta_2 << \delta_1,$$

$$\delta_2 << \delta_3 << \delta_1 \quad \text{και}$$
$$\delta_2 << \delta_1 << \delta_3.$$

Είναι εύκολο να διαπιστωθεί ότι και οι τρεις οδηγούν στην ίδια PDL – extension, την Th({conservative, radical, taxCut, spendingCut}).

Από τον ορισμό της PDL – extension E, E = In(Π) για μία κλειστή και επιτυχημένη διαδικασία Π. Οπότε:

Θεώρημα: Αν η E είναι μια PDL – extension μιας default θεωρίας με προτεραιότητες T = (W, D, <) τότε η E είναι και μια extension της (W, D).

Θεώρημα: Μια default θεωρία με προτεραιότητες T = (W, D, <) πάντα έχει μια PDL – extension αν το D είναι υπολογίσιμο.

Δυναμικές Προτεραιότητες:

Το μειονέκτημα των στατικών προτεραιοτήτων είναι θα πρέπει ο χρήστης να παρέχει όλη την πληροφορία των προτεραιοτήτων η οποία δεν θα αλλάζει ποτέ. Αυτό μπορεί να είναι πολύ περιοριστικό για κάποιες εφαρμογές. Μια άλλη προσέγγιση είναι αυτή στην οποία η πληροφορία προτεραιοτήτων είναι μέρος της γλώσσας της λογικής και μπορεί να χρησιμοποιηθεί όπως και κάθε άλλη δήλωση. Για παράδειγμα, είναι πιθανό να γράφουμε:

$$\frac{p : d_1 < d_2}{d_1 < d_2}$$

για να εκφράσουμε ότι: 'Στις περιπτώσεις που το p ισχύει, συνήθως το d_1 θα πρέπει να προτιμάται σε σχέση με το d_2'.Ασφαλώς, υπάρχουν extensions στις οποίες $d_1<d_2$ μπορεί να μην χρησιμοποιηθεί ως υπόθεση εξαιτίας άλλων δεδομένων που είναι διαθέσιμα.

Ο συλλογισμός με βάση τις προτεραιότητες είναι πολύ ευέλικτος, άλλα η μεγάλη εκφραστικότητα μπορεί να φέρει προβλήματα. Για παράδειγμα, είναι πιθανό ακόμα κι αν τα defaults είναι σε κανονική μορφή να μην υπάρχει extension. Αν είχαμε μια θεωρία που περιλαμβάνει τους παρακάτω default κανόνες:

$$d_1 = \frac{true : d_2 < d_1}{d_2 < d_1},\ d_2 = \frac{true : d_1 < d_2}{d_1 < d_2}.$$

Βλέπουμε ότι αυτή η θεωρία δεν έχει καμία extension.

Παρουσίαση Αλγορίθμου.

Παραδοχές:

Για την ανάπτυξη και υλοποίηση του αλγορίθμου χρησιμοποιήθηκε η γλώσσα λογικού προγραμματισμού Prolog. Για να ορίσουμε την άρνηση όχι λόγω αποτυχίας αλλά το γεγονός ότι είναι αληθές το not ενός τύπου δηλαδή τον τελεστή '$\neg$' χρησιμοποιούμε ένα κανόνα χωρίς σώμα της μορφής n(X) όπου X κάθε τύπος της λογικής πρώτης τάξης (π.χ. το $\neg p$ αναπαριστάται ως n(p)). Επιπλέον, για να εκφράσουμε τη σύζευξη δυο τύπων ή το πλήθος πολλών τύπων κάνουμε χρήση των λιστών είτε αυτά είναι στα προαπαιτούμενα είτε είναι στις αιτιολογίες είτε στα επακόλουθα και μετά τα χειριζόμαστε κατάλληλα ένα προς ένα τα μέλη της. Δεν υποστηρίζεται ότι οι κανόνες που έχουν πάνω από ένα προαπαιτούμενο ή αιτιολογία ή επακόλουθο χειρίζονται ως να πρέπει να ισχύει η σύζευξη τους απαραίτητα κάτι που θα ήταν πολύ περιοριστικό. Επιπρόσθετα, στην περίπτωση που έχουμε defaults με ελεύθερες μεταβλητές χρησιμοποιούμε μια τεχνική παρόμοια με την Skolemization με τη διαφορά ότι παίρνουμε ένα αντίγραφο του κανόνα κάθε φορά και σε αυτό κάνουμε instantiate.

Προτεραιότητες:

Υπάρχουν δυο τρόποι να ορίσουμε άμεσα την πληροφορία των προτεραιοτήτων: είτε με χρήση στατικών προτεραιοτήτων όπου συνδέω κάθε κανόνα με ένα επιπλέον χαρακτηριστικό τον αριθμό που δηλώνει την προτεραιότητα του είτε με χρήση δυναμικών. Στον αλγόριθμο χρησιμοποιείται μια μορφή 'δυναμικών' προτεραιοτήτων που ο χρήστης εισάγει με μορφή λίστας τις προτιμήσεις του απέναντι στους κανόνες. Ένας τρόπος ήταν να βρίσκουμε ποιοι κανόνες είναι εφαρμόσιμοι στην συγκεκριμένη κατάσταση και μετά ανάλογα με την προτίμηση τους να αποφασίζουμε ποιον θα εφαρμόσουμε. Ωστόσο, επειδή μια τέτοια διαδικασία θα ήταν χρονοβόρα υιοθετήθηκε η ιδέα ότι καλύτερα να κατατάσσω τους κανόνες σε φθίνουσα διάταξη με βάση την προτίμηση του χρήστη και να χρησιμοποιώ αυτόν που μπορεί να ενεργοποιηθεί και έχει την μεγαλύτερη προτεραιότητα.

Extensions:

Στην ουσία για τον υπολογισμό μιας extension ακολουθούμε τον αλγόριθμο του Αντωνίου ως ένα βαθμό με τη διαφορά ότι δεν απαιτείται να είναι οι διαδικασίες κλειστές για να τις αποδεχτούμε ως extensions. Το μόνο που απαιτείται είναι να είναι επιτυχής. Με αυτόν τον τρόπο κερδίζουμε επιπλέον νέα γνώση, όμως γίνονται και πολλοί έλεγχοι για την διασφάλιση εξαγωγής χρήσιμης και όχι εσφαλμένης πληροφορίας καθώς και συνέπειας στο τι θα ήταν ορθό να υποθέσουμε. Ωστόσο, οι διαδικασίες που βασίζονται τα extension (είναι το σύνολο In) που βρίσκουμε κάνουν και χρήση της λογικής του Prakken με την έννοια ότι μια διαδικασία αντιμετωπίζεται σαν να είναι ένα επιχείρημα. Δηλαδή, αφού υπολογίσουμε όλες τις extensions τις ομαδοποιούμε σε μια λίστα, τις συγκρίνουμε βρίσκουμε ποιες διαδικασίες τους από αυτές αποτελούν τα δικαιολογημένα επιχειρήματα και επομένως και ποιες αποτελούν επιχειρήματα που αποτρέπονται και μετά ποιες είναι τα defensible επιχειρήματα.

Ας μελετήσουμε, όμως , καλύτερα τον αλγόριθμο με την παρουσίαση του σε μορφή κανόνων και την επεξήγηση του.

Συνέπεια:

Ο αλγόριθμος έχει δημιουργηθεί με στόχο να αποφεύγονται οι ασυνεπείς καταστάσεις. Παρουσιάζει συνέπεια:

- Ως προς το συνδυασμό διαφόρων αντικρουόμενων αιτιολογιών των defaults.
- Ως προς το αν έχουμε κάποιο προηγούμενο αντιφατικό με το παρών συμπέρασμα.
- Με το αν πληρούνται οι προϋποθέσεις ισχύος των αιτιολογιών κατά την εφαρμογή ενός κανόνα.
- Με το να μην χρησιμοποιηθεί για δεύτερη φορά ένα default μιας και δεν θα προέκυπτε επιπλέον γνώση.
- Με το να μην υπάρχουν πουθενά διπλότυπα.

Ψευδοκώδικας:

1η Συνάρτηση:

//Υπολογισμός των extensions και ομαδοποίησης τους σε μια λίστα E

//Είσοδος από τον χρήστη:

1.Λίστα γεγονότων W
2.Λίστα από κανόνες defaults Din
3.Λίστα προτεραιοτήτων Pr

//Έξοδος προγράμματος:

1.Λίστα από extensions E

1ο Βήμα: Συσχέτιση κάθε default κανόνα με την προτεραιότητα που του δίνει ο χρήστης

//Οι default κανόνες που δίνονται στην είσοδο είναι της μορφής default(A,B,C)

//όπου το A: προαπαιτούμενο, το B: αιτιολογία και το C: επακόλουθο

//Μετά την εφαρμογή της συσχέτισης τους με τις προτεραιότητες δημιουργείται το σύνολο D
//που περιλαμβάνει default κανόνες της μορφής default(n, A, B, C) όπου το n>0 και
//δηλώνει την προτεραιότητα του κανόνα (όσο μεγαλύτερη τιμή τόσο μεγαλύτερη //προτίμηση)
2^{ο} **Βήμα:** Ενημέρωση του In συνόλου έτσι ώστε να περιέχει τα γεγονότα που βρίσκονται στο W.
3^{ο} **Βήμα:** Δημιουργία αντιγράφου της λίστας με τους Default κανόνες για να δουλεύει σωστά με ανοιχτούς κανόνες
4^{ο} **Βήμα:** Επιλογή του Default κανόνα με την μεγαλύτερη προτεραιότητα
Όταν υπάρχουν defaults με την ίδια προτεραιότητα, τότε επιλέγεται ένας στην τύχη και μετά όταν ολοκληρωθεί η διαδικασία επιστρέφουμε στην κατάσταση που την έχουμε αποθηκεύσει (δηλαδή το In, το PCur, το Out) και ξανατρέχουμε τα βήματα επιλέγοντας έναν άλλο από αυτούς με την ίδια προτεραιότητα και υπολογίζουμε τις άλλες διαδικασίες που προκύπτουν.
5^{ο} **Βήμα:** Έλεγχος αν τα prerequisites είναι έγκυρα , δηλαδή αν ανήκουν στο σύνολο In που ισχύει ως τώρα.
6^{ο} **Βήμα:** Έλεγχος αν μπορώ να υποθέσω ότι ισχύουν οι Conditions εφαρμογής του default.
//Δηλαδή αν δεν υπάρχει το n(B) ($\neg$ B) στο σύνολο In
7^{ο} **Βήμα:** Έλεγχος αν το συμπέρασμα είναι συνεπές, μην έχω αντικρουόμενα συμπεράσματα
//Δηλαδή αν το n(C) δεν ανήκει στο σύνολο In
8^{ο} **Βήμα:** Έλεγχος αν δεν έχει ξαναχρησιμοποιηθεί ο ίδιος κανόνας (ίδιο instance του κανόνα)
//Δηλαδή να μην είναι μέλος της διαδικασίας P.
9^{ο} **Βήμα:** Αν δεν αποτύχει ο παραπάνω έλεγχος: ενημέρωση της διαδικασίας με τον νέο κανόνα, αλλιώς: πηγαίνω στο 3^{ο} Βήμα.
10^{ο} **Βήμα:** Έλεγχος αν προκύπτει νέα γνώση, ενημέρωση του συνόλου In με το συμπέρασμα του κανόνα default εκτός κι αν ήδη υπάρχει οπότε και επιστρέφω στο 3^{ο} Βήμα
11^{ο} **Βήμα:** Ενημέρωση συνόλου Out.
//Εισάγουμε το n(B) στο σύνολο Out
12^{ο} **Βήμα:** Έλεγχος αν η διαδικασία ως εδώ είναι επιτυχής.
//Δηλαδή να ισχύει ότι $In \cup Out = \{\}$
13^{ο} **Βήμα:** Αποθήκευση του In της διαδικασίας που είναι και το extension στη λίστα E.
14^{ο} **Βήμα:** Αποθήκευση της διαδικασίας PCur που είναι η διαδικασία στη λίστα P.
15^{ο} **Βήμα:** Επιστροφή στο 3^{ο} Βήμα για την εφαρμογή κάποιου άλλου κανόνα μέχρι να μην υπάρχουν άλλοι κανόνες που να μπορούν να εφαρμοστούν
16^{ο} **Βήμα:** Ξανατρέχουμε την συνάρτηση οπισθοδρομώντας στην κατάσταση που είχαμε δύο κανόνες με ίδια προτεραιότητα από το 3^{ο} Βήμα για να δούμε αν υπάρχει κάποια άλλη διαδικασία που μπορεί να δημιουργηθεί

2^{η} Συνάρτηση:
//Χειρισμός των extensions της προηγούμενης συνάρτησης και κατηγοριοποίησή τους σε δικαιολογημένα, ανατρέποντα και αναιρέσιμα επιχειρήματα
//Είσοδος:

1.Λίστα από extensions E
που επιστρέφει η προηγούμενη συνάρτηση
2.Λίστα από τις διαδικασίες P
που οι extensions είναι τα σύνολα In
που σχετίζονται με αυτές

//Έξοδος προγράμματος:

1.Λίστα από δικαιολογημένα επιχειρήματα J
2.Λίστα από επιχειρήματα που ηττήθηκαν O
3.Λίστα από αναιρέσιμα επιχειρήματα Ds.

1° Βήμα: Συγκρίνοντας τις extensions βρίσκουμε αυτές που είναι υποσύνολα των άλλων ή και ίδιες και συγκρίνοντας τα defaults που περιλαμβάνουνε οι διαδικασίες τους υπολογίζουμε τα δικαιολογημένα επιχειρήματα και τα ομαδοποιούμε στη λίστα J καθώς και αυτά που νικιούνται και τα ομαδοποιούμε στη λίστα O.
//Η λογική είναι η εξής: Μια διαδικασία είναι δικαιολογημένο επιχείρημα αν τα defaults που περιλαμβάνει είναι το μικρότερο σύνολο τέτοιο ώστε να βγάζουμε τα περισσότερα συμπεράσματα (μέγιστη extension) και επιπλέον να έχουν οι κανόνες της τις μεγαλύτερες προτεραιότητες.
2° Βήμα: Στο προηγούμενο βήμα όσες extensions και διαδικασίες δεν συμμετείχαν στην σύγκριση ή δεν ηττήθηκαν αλλά και ούτε ανήκουν στα δικαιολογημένα επιχειρήματα γιατί δεν μπορούμε να αποφανθούμε ότι οι κανόνες της διαδικασίας τους έχουν τις μεγαλύτερες προτεραιότητες από άλλες με τα ίδια συμπεράσματα ή τα συμπεράσματα τους είναι διαφορετικά ονομάζονται αναιρέσιμα (defensible) επιχειρήματα και ομαδοποιούνται στη λίστα Ds.

Παραδείγματα Κατανόησης Αλγορίθμου:

1. Έστω ότι έχουμε μια θεωρία T = (W,D,P) όπου

W = friend (tom, bob), friend (bob, sally), friend (sally, tina)

(τα γεγονότα είναι: ο tom και ο bob είναι φίλοι, ο bob και η sally είναι φίλοι, και η sally και η tina είναι φίλες)

D = default (friend(X, Y) $\wedge$ friend(Y, Z): friend(X, Z) /friend(X, Z))

(το default ουσιαστικά λέει αν οι φίλοι των φίλων μου είναι συνήθως και δικοί μου φίλοι)

P = 1 (αφού είναι ένας μόνο κανόνας).

Μετά την εκτέλεση του αλγορίθμου μου καταλήγουμε στα εξής:
Επεκτάσεις(Extensions):
1. [friend(tom, bob), friend(bob, sally), friend(sally, tina), friend(tom, sally), friend(tom, tina), friend(bob, tina)]

Διαδικασίες:
Δεν υπάρχουν δικαιολογημένα επιχειρήματα καθώς δεν παρέχεται κάποια πληροφορία για το αν κάποιος κανόνας νικά κάποιον άλλον. Έτσι, έχουμε τα αναιρέσιμα επιχειρήματα:

1. $[\frac{friend(tom,bob) \wedge friend(bob,sally) : friend(tom,sally)}{friend(tom,sally)},$
$\frac{friend(tom,sally) \wedge friend(sally,tina) : friend(tom,tina)}{friend(tom,tina)},$
$\frac{friend(bob,sally) \wedge friend(sally,tina) : friend(bob,tina)}{friend(bob,tina)}]$
2. $[\frac{friend(tom,bob) \wedge friend(bob,sally) : friend(tom,sally)}{friend(tom,sally)},$,
$\frac{friend(bob,sally) \wedge friend(sally,tina) : friend(bob,tina)}{friend(bob,tina)},$
$\frac{friend(tom,bob) \wedge friend(bob,tina) : friend(tom,tina)}{friend(tom,tina)}]$

Αν είχαμε την πληροφορία ότι $\frac{friend(tom,bob) \wedge friend(bob,tina) : friend(tom,tina)}{friend(tom,tina)} >$ $\frac{friend(tom,sally) \wedge friend(sally,tina) : friend(tom,tina)}{friend(tom,tina)}$ τότε το 2^o αναιρέσιμο επιχείρημα θα ήταν δικαιολογημένο και το 1^o θα άνηκε σ' αυτά που ηττήθηκαν.

Όλα τα υποσύνολα της επέκτασης 1 μας δίνουν οι διαδικασίες τους το σύνολο των ηττημένων επιχειρημάτων.

2. Έστω ότι έχουμε μια θεωρία με W = [p, q], D που περιλαμβάνει τους κανόνες: δ_1 = [q] : w / w και δ_2 = p : n (w) / r και ισχύει ότι $\delta_1 << \delta_2$ τότε ο αλγόριθμός μας θα μας δώσει:
Επέκταση(extension):
1. [w, p, q]
Διαδικασίες:
Δικαιολογημένο επιχείρημα:
1. $[\frac{[q]:w}{w}]$
Ηττημένο, ανατρεπόμενο επιχείρημα:
1. $[\frac{p:n(w)}{r}]$

3. Έστω ότι έχουμε τη θεωρία με W = [green, aaaMember] (όπου τα γεγονότα είναι ότι του κάποιου του αρέσει το πράσινο και είναι aaaMember) και με D = [default(green, n (likeCars), n(likeCars)), default (aaaMember, likeCars, likeCars)] (όπου τα defaults λένε ότι αν κάποιου του αρέσει το πράσινο και μπορούμε να υποθέσουμε ότι δεν του αρέσουν τα αμάξια τότε υποθέτουμε ότι δεν του αρέσουν όντως και ότι αν κάποιος είναι aaaMember και είναι συνεπές με την παρούσα γνώση να συμπεράνουμε ότι του αρέσουν τα αμάξια το συμπεραίνουμε)και με P []. Τότε ο αλγόριθμος μας δίνει:
Επέκταση(extension):
1. [green, aaaMember, n(likeCars)]
2. [green, aaaMember, likeCars]
Διαδικασίες:
Αναιρέσιμα επιχειρήματα:
1. $[\frac{green:n(likeCars)}{n(likeCars)}]$
2. $[\frac{aaaMember:likeCars}{likeCars}]$
Τα επιχειρήματα είναι αναιρέσιμα μιας και δεν υπάρχει ένας τρόπος να προτιμήσουμε κάποιο από τα δύο αφού δεν υπάρχουν προτεραιότητες.

Κεφάλαιο 6ο
Επεκτάσεις – Συμπεράσματα – Μελλοντικοί Στόχοι.

Δυσκολίες.

Οι δυσκολίες που συνάντησα κατά την υλοποίηση του αλγορίθμου ήταν αρκετές. Καταρχήν, δεν υπάρχουν ακόμα πολλές δημοσιεύσεις γύρω από το θέμα ούτε και μια σαφώς καθορισμένη θεωρία που θα σου δίνει πληροφορίες για το αν είναι σωστό να δίνουμε περισσότερη σημασία στην συνέπεια παρά στην ποσότητα της νέας πληροφόρησης. Σε μερικές περιπτώσεις ακόμα και αντιφατικές υποθέσεις μπορεί να είναι ορθό να γίνουν αν δεν οδηγούν σε αντικρουόμενα συμπεράσματα και σε μελλοντικά προβλήματα εσφαλμένης γνώσης.

Επιπλέον, απαιτείται ιδιαίτερη προσοχή σε περιπτώσεις που οι default κανόνες περιέχουν ελεύθερες μεταβλητές καθώς μπορεί η μη σωστή χρήση τους να μας οδηγήσει σε τελείως λάθος αποτελέσματα. Απαιτείται η χρήση αντιγράφου στην περίπτωση της Prolog για να μπορέσουμε να τα χειριστούμε ορθά σε προγράμματα παρόμοια με το δικό μου. Η έκφραση, στην συνέχεια του τελεστή $\neg$ με την μορφή κανόνα χωρίς σώμα όπως υλοποιήθηκε απαιτεί προσεκτικό χειρισμό καθώς η αγνόηση του σε κάποιο σημείο είναι καταστροφική για την εξαγωγή ορθών συμπερασμάτων.

Επεκτάσεις.

Ο αλγόριθμος που παρουσιάστηκε αν και αντιμετωπίζει ικανοποιητικά ένα πλήθος από default λογικές ωστόσο χρειάζεται κάποιες επεκτάσεις για να αντιμετωπίζει default θεωρίες που στους τύπους τους έχουν διάζευξη δεδομένων. Επιπλέον, θα μπορούσε να επεκταθεί και να συμπεριλάβει στη γλώσσα του τις προτεραιότητες έτσι ώστε να τις χρησιμοποιεί με δυναμικό τρόπο και μόνο όπου χρειάζεται. Δηλαδή οι προτεραιότητες να αποτελούν ουσιαστικά μια μορφή κανόνων που θα πρέπει να εφαρμόζονται όταν υπάρχει σύγκρουση ή όταν εφαρμόζονται σε μια κατάσταση πάνω από ένα default κανόνες. Ακόμη, θέτει αρκετούς περιορισμούς όσον αφορά την αποφυγή σε κατάληξη ασυνεπών καταστάσεων που περιορίζει όμως το πλήθος των αποτελεσμάτων του. Τέλος, θα μπορούσε να ενσωματώσει και μια δεοντική λογική σε συνδυασμό με την default και να δημιουργηθεί ένα πλαίσιο εργασίας περισσότερο εύχρηστο σε σχέση με αυτό του Prakken. Θα μπορούσε δηλαδή να υπάρχει κάποιος κατάλληλος χειρισμός των defaults και των δεδομένων που περιλαμβάνουν δεοντικούς τελεστές (O, P , F). Στην Prolog θα μπορούσε να επιτευχθεί με την προσθήκη κανόνων για το χειρισμό αυτών των καταστάσεων.

Συμπεράσματα.

Η Default Συλλογιστική είναι μια σημαντική μέθοδος αναπαράστασης γνώσης και συλλογισμού, επειδή υποστηρίζει την εξαγωγή συμπερασμάτων με ελλιπής γνώση και επειδή defaults μπορούν να βρεθούν σε πολλά πεδία εφαρμογών, όπως διαγνωστικά προβλήματα, κανονισμοί, ανάκτηση πληροφοριών και εξόρυξη νέας γνώσης, νομικό συλλογισμό. Η Default Λογική μπορεί να χρησιμοποιηθεί είτε για να μοντελοποιήσει το συλλογισμό με βάση ελλιπή δεδομένα ή για την συμπαγή και οργανωμένη παρουσίαση των δεδομένων.

Διάφορες παραλλαγές έχουν αναπτυχθεί, είτε για να υπερνικήσουν κάποιες ανεπάρκειες του πρωτότυπου συλλογισμού είτε για να εισάγουν νέες πλευρές του με την χρήση αποτελεσματικών μεθόδων εξαγωγής όχι ακριβώς συμπερασμάτων άλλα αντιλήψεων.

Μελλοντικοί Στόχοι.

Ο αλγόριθμος αυτός θα μπορούσε να επεκταθεί σε ένα διαδικτυακό υπολογιστικό περιβάλλον υλοποίησης μη μονότονης συλλογιστικής με τη χρήση της γλώσσα Jess και την εισαγωγή και εξαγωγή (αλληλεπίδραση) δεδομένων από το χρήστη με την βοήθεια της XML γλώσσας. Ένα πρότυπο για το πώς θα μπορούσαν να οριστούν τα δεδομένα εισαγωγής του χρήστη με XML είναι:

DTD Αρχειο:

```
<!DOCTYPE input_process [
        <!ELEMENT input_process (facts,defaults)>
        <!ELEMENT facts (fact*)>
        <!ELEMENT fact (#PCDATA)>
        <!ELEMENT defaults (default+)>
        <!ELEMENT default (prerequisite,justifications,conclusion)>
        <!ELEMENT prerequisite (#PCDATA)>
        <!ELEMENT justifications (justification+)>
        <!ELEMENT justification (#PCDATA)>
        <!ELEMENT conclusion (#PCDATA)>
]>
```

XML Αρχείο:

input_process.xml

```
<?xml version="1.0"?>
<!DOCTYPE input_process SYSTEM "input_process.dtd">
<input_process>
    <facts>
        <fact>p</fact>
        <fact>q</fact>
    </facts>
    <defaults>
        <default>
            <prerequisite>p</prerequisite>
            <justifications>
                <justification>p</justification>
            </justifications>
            <conclusion>w</conclusion>
        </default>
        <default>
            <prerequisite>q</prerequisite>
            <justifications>
                <justification>q</justification>
            </justifications>
            <conclusion>r</conclusion>
        </default>
    </defaults>
</input_process>
```

XML Schema

```
<?xml version="1.0"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
            targetNamespace=""
            xmlns=""
            elementFormDefault="qualified">
  <xsd:element name="input_process">
     <xsd:complexType>
      <xsd:sequence>
         <xsd:element ref="facts" minOccurs="1" maxOccurs="1"/>
           <xsd:element ref="defaults" minOccurs="1" maxOccurs="1"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="facts">
     <xsd:complexType>
      <xsd:sequence>
         <xsd:element ref="fact" minOccurs="1" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="fact" type="xsd:string"/>
  <xsd:element name="defaults">
     <xsd:complexType>
      <xsd:sequence>
         <xsd:element ref="default" minOccurs="1" maxOccurs="unbounded"/>
       </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="default">
    <xsd:complexType>
       <xsd:sequence>
         <xsd:element ref="prerequisite" minOccurs="1" maxOccurs="1"/>
         <xsd:element ref="justifications" minOccurs="1" maxOccurs="1"/>
         <xsd:element ref="conclusion" minOccurs="1" maxOccurs="1"/>
       </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="prerequisite" type="xsd:string"/>
  <xsd:element name="justifications">
     <xsd:complexType>
      <xsd:sequence>
         <xsd:element ref="justification" minOccurs="1" maxOccurs="unbounded"/>
       </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="justification" type="xsd:string"/>
  <xsd:element name="conclusion" type="xsd:string"/>
```

Παράρτημα
Εισαγωγικές Έννοιες

Προτασιακή Λογική.

Η *προτασιακή λογική (propositional logic)* αποτελεί την απλούστερη μορφή μαθηματικής λογικής. Στην προτασιακή λογική κάθε γεγονός του πραγματικού κόσμου αναπαρίσταται με μια λογική πρόταση, η οποία χαρακτηρίζεται είτε ως αληθής (t-true) ή ως ψευδής (f-false), μπορεί δηλαδή να έχει δυο λογικές τιμές. Οι λογικές προτάσεις συνήθως συμβολίζονται με λατινικούς χαρακτήρες P,Q,R, και ονομάζονται άτομα (atoms). Τα άτομα μπορούν να συνδυαστούν με τη χρήση λογικών συμβόλων ή συνδετικών (connectives) και οι σύνθετες προτάσεις που προκύπτουν ονομάζονται ορθά δομημένοι τύποι (well formed formulae). Τα συνδετικά της προτασιακής λογικής είναι:

Σύμβολο	**Ονομασία / Επεξήγηση**
$\wedge$	σύζευξη (Λογικό "ΚΑΙ")
$\vee$	διάζευξη (Λογικό "Η")
$\neg$	άρνηση
$\rightarrow$	συνεπαγωγή ("ΕΑΝ ΤΟΤΕ")
$\leftrightarrow$	διπλή συνεπαγωγή ή ισοδυναμία ("ΑΝ ΚΑΙ ΜΟΝΟ ΑΝ")

Παράδειγμα:

P: "Ο Νίκος είναι προγραμματιστής"
Q: "Ο Νίκος έχει Υπολογιστή"
$P \rightarrow Q$: *Εάν* "Ο Νίκος είναι προγραμματιστής", *τότε* "Ο Νίκος έχει Υπολογιστή"

Μια *ερμηνεία* (interpretation) I αντιστοιχεί τιμές αληθείας στα άτομα και επεκτείνεται σε σύνθετους τύπους με τη χρήση του παρακάτω πίνακα αληθείας (truth table) για να χειριστεί τους συνδέσμους:

P	Q	$\neg$P	$P \vee Q$	$P \wedge Q$	$P \rightarrow Q$	$P \leftrightarrow Q$
t	t	f	t	t	t	t
t	f	f	t	f	f	f
f	t	t	t	f	t	f
f	f	t	f	f	t	t

Στην περίπτωση που κάποιος τύπος (π.χ. $P \rightarrow Q$) είναι αληθής σύμφωνα με κάποια ερμηνεία τότε ο τύπος ικανοποιείται από την ερμηνεία και η ερμηνεία αποτελεί ένα μοντέλο του τύπου. Υπάρχουν οι εξής περιπτώσεις τύπων που επεκτείνονται και σε σύνολα ατομικών τύπων:

- *Ταυτολογία* (tautology) είναι ένας τύπος που είναι αληθής κάτω από οποιαδήποτε ερμηνεία. Αν ένας τύπος F είναι ταυτολογία γράφεται $\models F$.
- *Αντίφαση* (contradiction) είναι ένας τύπος που είναι ψευδής κάτω από οποιαδήποτε ερμηνεία.
- Ένας τύπος Q *συνεπάγεται λογικά* (implication) από τον τύπο P εάν κάθε μοντέλο του P είναι και μοντέλο του Q. Η περίπτωση αυτή συμβολίζεται $P \models Q$.

@ Δυο τύποι P, Q είναι *ισοδύναμοι* (equivalent) εάν οι πίνακες αληθείας τους είναι ίδιοι κάτω από οποιαδήποτε ερμηνεία. Η λογική ισοδυναμία ορίζεται με το σύμβολο “$\Leftrightarrow$”.

Λογικές Ισοδυναμίες:

Υπάρχει ένα σύνολο ισοδυναμιών που χρησιμοποιείται και είναι χρήσιμο κατά τη διαδικασία εξαγωγής συμπερασμάτων και είναι το εξής:

Ισοδυναμία	**Ονομασία**
$P \Leftrightarrow \neg\neg P$	Νόμος της διπλής άρνησης
$(\neg P \vee \neg Q) \Leftrightarrow \neg(P \wedge Q)$	Νόμος de Morgan
$(\neg P \wedge \neg Q) \Leftrightarrow \neg(P \vee Q)$	Νόμος de Morgan
$(P \vee Q) \wedge R \Leftrightarrow (P \wedge R) \vee (Q \wedge R)$	Επιμερισμός ως προς τη σύζευξη
$(P \wedge Q) \vee R \Leftrightarrow (P \vee R) \wedge (Q \vee R)$	Επιμερισμός ως προς τη διάζευξη
$(P \rightarrow Q) \Leftrightarrow \neg P \vee Q$	
$(P \leftrightarrow Q) \Leftrightarrow (P \rightarrow Q) \wedge (Q \rightarrow P)$	

Κανονικές Μορφές:

Οι κανονικές μορφές (canonical forms) είναι μορφές τύπων της λογικής στις οποίες δεν εμφανίζονται καθόλου κάποια συνδετικά και ακολουθούν μια συγκεκριμένη δομή. Κλασσικά παραδείγματα είναι η διαζευκτική και η συζευκτική μορφή της λογικής που χρησιμοποιούνται μόνο τα συνδετικά της διάζευξης, της σύζευξης και η άρνηση.

Κατηγορηματική Λογική ή Λογική Πρώτης Τάξης.

Η προτασιακή λογική υπονοεί ότι ο κόσμος αποτελείται από γεγονότα που είναι αληθή ή ψευδή, χωρίς καμιά δυνατότητα διαχωρισμού και προσπέλασης των οντοτήτων του κόσμου στα οποία αναφέρεται το συγκεκριμένο γεγονός. Η κατηγορηματική λογική (predicate logic) αποτελεί επέκταση της προτασιακής λογικής και λύνει το παραπάνω πρόβλημα. Σ’ αυτήν ο κόσμος περιγράφεται σαν ένα σύνολο αντικειμένων, ιδιοτήτων και σχέσεων που προσδίδονται σε αυτά.

Η κατηγορηματική λογική επεκτείνει την προτασιακή λογική εισάγοντας όρους (terms), κατηγορήματα(predicates) και ποσοδείκτες (quantifiers). Το σύνολο των συμβόλων (αλφάβητο) της κατηγορηματικής λογικής περιέχει:

- *Σταθερές* οι οποίες ξεκινούν με πεζά γράμματα ή αριθμούς.
- *Συναρτησιακά σύμβολα* όπως f, g, father-of… Σε κάθε σύμβολο συνάρτησης αντιστοιχεί και ένα αριθμός που ονομάζεται τάξη (arity). Η τάξη είναι το πλήθος των ορισμάτων ή παραμέτρων της συνάρτησης.
- *Μεταβλητές* των οποίων τα ονόματα ξεκινούν με κεφαλαίο γράμμα του λατινικού αλφαβήτου.
- *Σύμβολα κατηγορημάτων* τα οποία και αυτά έχουν τάξη.
- *Συνδετικά*. Είναι τα ίδια με αυτά της προτασιακής λογικής.
- *Ποσοδείκτες*. Ο υπαρξιακός “$\exists$” και ο καθολικός “$\forall$”.
- *Σύμβολα στίξης*. “(”, “)”, “,”.
- *Σύμβολα αληθείας*. t (αληθής), f (ψευδής).

Ένας *όρος* της κατηγορηματικής λογικής μπορεί να είναι είτε μια σταθερά (όπως τίγρης, tweety), είτε μια μεταβλητή (όπως X, Χρώμα), ή ένας συναρτησιακός όρος της μορφής $f(t_1,t_2,\ldots,t_n)$, όπου f είναι συναρτησιακό σύμβολο τάξης n και $t_1,t_2,\ldots,t_n$ είναι όροι. Ένα παράδειγμα συναρτησιακού όρου είναι:

εργαζόμενος(γιάννης, επάγγελμα(καθηγητής)).

Ένας ατομικός τύπος (atomic formulae) έχει την μορφή:

$p(a_1,a_2,\ldots,a_n)$

όπου το *p ονομάζεται κατηγόρημα* (*predicate*) και τα $a_1,a_2,\ldots,a_n$ *ορίσματα* (*arguments*) και απεικονίζει μια σχέση ανάμεσα σε μια διατεταγμένη πλειάδα (tuple) αντικειμένων στα οποία αναφέρονται τα ορίσματά του, και μπορεί να είναι αληθής, όταν η σχέση ισχύει ή ψευδής στην αντίθετη περίπτωση.

Παράδειγμα:

Κάθε άνθρωπος έχει όνομα $\forall x \exists y$ (άνθρωπος(X) $\rightarrow$ όνομα(X,Y)).

Βασικός όρος ή τύπος (ground term) είναι ένας όρος που δεν περιέχει καμία μεταβλητή, π.χ. προγραμματιστής(νίκος). Υπάρχουν, όμως και τύποι που δεν είναι βασικοί και η χρήση μεταβλητών σ' αυτούς κάνει απαραίτητη την εισαγωγή δύο εννοιών: την αντικατάσταση (substitution) και την ενοποίηση (unification).

Η *αντικατάσταση* αφορά απλά την αντικατάσταση των μεταβλητών που εμφανίζονται σ' έναν τύπο από κάποιους όρους. Μια αντικατάσταση παριστάνεται με $\{X_i/t_i\}$ όπου X_i η μεταβλητή που θα αντικατασταθεί από τον όρο t_i.

Παράδειγμα:

Η αντικατάσταση {X/δελφίνι} στον τύπο είναι(X, ψάρι) θα δώσει τον τύπο είναι(δελφίνι, ψάρι)

Η *ενοποίηση* είναι η διαδικασία κατά την οποία δύο εκφράσεις γίνονται συντακτικά όμοιες με τη χρήση αντικαταστάσεων.

Ο γενικότερος ταυτοποιητής δύο όρων, είναι εκείνος ο ενοποιητής που μετατρέπει το κοινό στιγμιότυπο (instance) των δύο όρων, στη γενικότερή του μορφή.

Παράδειγμα:

Οι ακόλουθες προτάσεις:

είναι(λιοντάρι, θηλαστικό, X) και
είναι(λιοντάρι,Y,σαρκοβόρο)

ενοποιούνται με την αντικατάσταση θ={X/σαρκοβόρο, Y/θηλαστικό}.

Ο *αναδρομικός αλγόριθμος για την εύρεση του γενικότερου ενοποιητή* (αυτού με τις λιγότερες αντικαταστάσεις) που χρησιμοποιεί η Prolog είναι:

1. Δυο σταθερές ενοποιούνται αν και μόνο αν είναι ίδιες.
2. Μια μεταβλητή ενοποιείται με οποιοδήποτε όρο, εισάγοντας μια νέα αντικατάσταση στο γενικότερο ταυτοποιητή.
3. Δυο συναρτησιακοί όροι ενοποιούνται αν έχουν το ίδιο συναρτησιακό σύμβολο, την ίδια τάξη κι αν κάθε όρισμα του πρώτου όρου μπορεί να ενοποιηθεί με το αντίστοιχο σε θέση όρισμα του δεύτερου συναρτησιακού όρου.
4. Δυο ατομικοί τύποι ενοποιούνται αν έχουν το ίδιο κατηγόρημα, την ίδια τάξη κι αν κάθε όρισμα του πρώτου μπορεί να ενοποιηθεί με το αντίστοιχο σε θέση όρισμα του δεύτερου ατομικού τύπου.

Έστω ένα πεδίο D που είναι ένα μη κενό σύνολο στοιχείων. Μια *ερμηνεία* I πάνω στο D είναι μια απεικόνιση στην οποία:

- Κάθε σταθερά αντιστοιχείται σε ένα στοιχείο του D.
- Κάθε μεταβλητή έχει σαν πεδίο τιμών το D.
- Κάθε συναρτησιακό σύμβολο f τάξης n ορίζεται σα μια απεικόνιση από το χώρο D' στο D.
- Κάθε κατηγόρημα τάξης n ορίζεται σα μια απεικόνιση από το χώρο D' στα σύμβολα αληθείας {t,f}. Ουσιαστικά η ερμηνεία ορίζει ένα υποσύνολο D' για το οποίο το κατηγόρημα είναι αληθές.

Κλειστός τύπος είναι ο τύπος που περιέχει μόνο ποσοτικοποιημένες μεταβλητές και η λογική τιμή του μπορεί να υπολογιστεί μόνο βάση μιας ερμηνείας.

Λογικές Ισοδυναμίες:

Στην κατηγορηματική λογική υπάρχουν οι ισοδυναμίες της προτασιακής λογικής και επιπλέον κάποιες ισοδυναμίες που αφορούν τους ποσοδείκτες και είναι οι εξής:

Ισοδυναμία	Ονομασία
$\forall X(p(X)) \Leftrightarrow \neg\exists X \neg p(X)$	
$\exists X(p(X)) \Leftrightarrow \neg\forall X \neg p(X)$	
$\forall X(p(X)) \vee q \Leftrightarrow \forall X(p(X) \vee q)$	όπου το q δεν περιέχει ελεύθερες εμφανίσεις της μεταβλητής X
$\forall X(p(X)) \wedge q \Leftrightarrow \forall X(p(X) \wedge q)$	
$\exists X(p(X)) \vee q \Leftrightarrow \exists X(p(X) \vee q)$	
$\exists X(p(X)) \wedge q \Leftrightarrow \exists X(p(X) \wedge q)$	
$\forall X(p(X)) \Leftrightarrow \forall Y(p(Y))$	
$\exists X(p(X)) \Leftrightarrow \exists Y(p(Y))$	
$\forall X(p(X)) \wedge \forall X(q(X)) \Leftrightarrow \forall X(p(X) \wedge q(X))$	
$\exists X(p(X)) \wedge \exists X(q(X)) \Leftrightarrow \exists X(p(X) \wedge q(X))$	

Δυνατότητα Υπολογισμού:

Έστω ένα πρόβλημα Π. Υποθέτουμε ότι υπάρχει ένας αλγόριθμος ο οποίος μπορεί να δίνει σωστές απαντήσεις, σε κάθε στιγμιότυπο (instance) του Π. Τότε, το πρόβλημα Π θα ονομάζεται *υπολογίσιμο* (*computable*). Μια ειδική κλάση προβλημάτων, είναι αυτή που η απάντηση του αλγορίθμου σε κάθε στιγμιότυπο ενός προβλήματος μπορεί να είναι είτε "ναι"' είτε "όχι"'. Εάν ένα πρόβλημα από αυτή την κλάση είναι υπολογίσιμο, θα ονομάζεται *ικανό απόφασης* (*decidable*) και ο αλγόριθμος που το επιλύει *διαδικασία απόφασης* (*decision procedure).* Αν ένα τέτοιο πρόβλημα δεν είναι υπολογίσιμο, θα ονομάζεται *μη ικανό απόφασης* (*undecidable*).

Αποδεικτικές Διαδικασίες:

Ονομάζουμε αποδεικτική διαδικασία κάθε μέθοδο που χρησιμοποιείται για ν' αποδείξουμε ότι ένας τύπος φ είναι λογική συνέπεια κάποιου συνόλου τύπων Σ. Συνήθως, οι τύποι στο Σ ονομάζονται υποθέσεις (premises) και ο φ συμπέρασμα (conclusion) της απόδειξης.

Μια απόδειξη αποτελείται από ένα ή περισσότερα βήματα, σε καθένα από τα οποία παράγεται ένας νέος τύπος από τις υποθέσεις και τους προηγούμενα παραγμένους τύπους. Κάθε βήμα της απόδειξης βασίζεται σε κάποιους κανόνες παραγωγής (derivation rules). Μια διαδικασία απόδειξης που συνήθως χρησιμοποιείται στα μαθηματικά, είναι η *modus ponens*. Σύμφωνα με τη διαδικασία αυτή, από το σύνολο των τύπων {φ, (φ→ ψ)} μπορούμε να πάρουμε τον τύπο ψ. Μπορούμε να συνδέσουμε πολλούς κανόνες modus ponens μαζί, ώστε να έχουμε μια απόδειξη. Αν ο τύπος φ μπορεί να παραχθεί από το Σ με χρήση κάποιων κανόνων παραγωγής, λέμε ότι ο φ είναι *θεώρημα* του υπολογισμού από το Σ και των συγκεκριμένων κανόνων παραγωγής.

Η *ορθότητα* (*soundness*) και η *πληρότητα* (*completeness*) είναι δυο επιθυμητές ιδιότητες των αποδεικτικών διαδικασιών. Μια αποδεικτική διαδικασία ονομάζεται *ορθή* (*sound*), όταν επιτρέπει μόνο λογικές συνέπειες να παράγονται από τις υποθέσεις. Ονομάζεται *πλήρης* (*complete*), όταν είναι ικανή να παράγει όλες τις λογικές συνέπειες των υποθέσεων. Ο modus ponens είναι ορθός, αλλά όχι πλήρης.

Μορφή Kowalski:

Μια μορφή εκφράσεων της προτασιακής λογικής με ιδιαίτερο ενδιαφέρον είναι η μορφή Kowalski. Σ' αυτήν όλες οι προτάσεις εκφράζονται σαν λογικές ισοδυναμίες της μορφής:

$$q_1, q_2, \ldots, q_n \rightarrow r_1, r_2, \ldots, r_n$$

Οι ατομικοί τύποι r_i στην παραπάνω έκφραση είναι σε διάζευξη και αποτελούν τα συμπεράσματα της πρότασης, ενώ οι q_i σε σύζευξη και αποτελούν τις υποθέσεις.

Προτάσεις Horn:

Μια ειδική περίπτωση προτάσεων της προτασιακής λογικής με μεγάλη σημασία σε πρακτικά συστήματα όπως στο λογικό προγραμματισμό με Prolog είναι οι προτάσεις Horn. Στις προτάσεις Horn, επιτρέπεται μόνο ένας ατομικός τύπος στο συμπέρασμα, είναι δηλαδή της μορφής:

$$q_1, q_2, \ldots, q_n \rightarrow r$$

Πλεονεκτήματα και Μειονεκτήματα Κατηγορηματικής Λογικής:

Πλεονεκτήματα:

- αντιστοιχία με τη φυσική γλώσσα,
- η ικανοποιητική έκφραση ποσοτικοποίησης των εννοιών με τους κατάλληλους ποσοδείκτες και
- η ικανότητά της να συλλάβει τη γενικότητα.

Μειονεκτήματα:

- αδυναμία έκφρασης της ασάφειας,
- η αθροιστικότητα των αποτελεσμάτων και
- δεν προσφέρει τη δυνατότητα λογισμού με εύλογες υποθέσεις.

Λογικός Προγραμματισμός:

Η χρήση των κανονικών προγραμμάτων ως μέσο αναπαράστασης γνώσης αλλά και για υπολογισμούς, υλοποιείται μέσω του λογικού προγραμματισμού. Σύμφωνα με τον ορισμό ενός κανονικού προγράμματος, ένα κανονικό λογικό πρόγραμμα είναι ένα πεπερασμένο σύνολο κανονικών προτάσεων. Κάθε λεκτικό στοιχείο που εμφανίζεται σ' ένα στόχο ή στο σώμα μιας πρότασης θα ονομάζεται *υποστόχος* (*subgoal*).

Σύνταξη:

Η σύνταξη των λογικών προγραμμάτων ακολουθεί τη σύνταξη της κατηγορηματικής λογικής πρώτης τάξης.

- Στα λογικά προγράμματα, το σύμβολο της συνεπαγωγής '$\leftarrow$' μιας πρότασης Horn γράφεται ':-', ένα αρνητικό λεκτικό στοιχείο $\neg A$ στο σώμα μιας πρότασης θα γράφεται 'not *A*', ενώ τα σύμβολα $\vee$(ή), $\wedge$(και) είναι τα ';', ',' αντίστοιχα.
- Μια πρόταση έχει τη μορφή A :- B_1, B_2, …, B_λ όπου $\lambda \geq 0$. Το *A* ονομάζεται *κεφαλή* (*head*) της πρότασης και το τμήμα B_1, B_2, …, B_λ *σώμα* (*body*). Αν $\lambda \geq 0$ η πρόταση ονομάζεται *κανόνας* (*rule*), ενώ αν $\lambda = 0$ ονομάζεται *γεγονός* (*fact*) (ή *μοναδιαία πρόταση –unit clause*) και γράφεται, απλά, *A*. Ένας στόχος (η ερώτημα - query) σ' ένα λογικό πρόγραμμα,

εκφράζεται από το σύμβολο '?-' ακολουθούμενο από μια σύζευξη λεκτικών στοιχείων. Κάθε πρόταση ή στόχος τερματίζεται από μια τελεία '.'.

- Τα ονόματα των κατηγορημάτων είναι ακολουθίες συμβόλων (strings) και αρχίζουν με πεζό γράμμα του λατινικού αλφαβήτου.
- Μια συλλογή προτάσεων που έχουν το ίδιο όνομα κατηγορήματος για κεφαλή ορίζει μια σχέση (relation) και ονομάζεται *διαδικασία* (procedure).
- Η βασική δομή στα λογικά προγράμματα είναι ο *όρος* (*term*). Ένας όρος μπορεί να είναι μια σταθερά (ακέραιος αριθμός ή άτομο), μια μεταβλητή ή ένας σύνθετος όρος. Τα άτομα τα συμβολίζουμε με ακολουθίες χαρακτήρων που αρχίζουν με πεζά γράμματα του λατινικού αλφαβήτου ή με ακολουθίες οποιονδήποτε άλλων συμβόλων περικλειόμενες από απλά εισαγωγικά.
- Ένας *σύνθετος όρος* (*compound term*) είναι μια δομή η οποία συγκροτείται από ένα σύμβολο συνάρτησης (που ονομάζεται και *συναρτησιακός τελεστής -functor)* και μια ακολουθία ενός η περισσότερων όρων που ονομάζονται *ορίσματα* (*arguments)*. Ένας σύνθετος όρος χαρακτηρίζεται από το όνομά του, το οποίο είναι ένα άτομο, και από τον αριθμό των ορισμάτων του που ονομάζεται *τάξη* (*arity*). Ένας συναρτησιακός τελεστής τάξης *k* θα συμβολίζεται *f*/*k*. Ένας όρος που δεν περιλαμβάνει καμία μεταβλητή θα ονομάζεται *βασικός* (*ground*).
- Τα ονόματα των μεταβλητών αρχίζουν με κεφαλαίο γράμμα ή από το σύμβολο '_'. Μεταβλητές που εμφανίζονται μόνο μία φορά σε μια πρόταση δεν χρειάζεται να ονομάζονται. Τέτοιες μεταβλητές ονομάζονται *ανώνυμες* (*anonymous)* και συμβολίζονται με το σύμβολο '_'. Διακεκριμένες εμφανίσεις ανώνυμων μεταβλητών σε μια πρόταση θεωρούνται διαφορετικές μεταξύ τους.

Λογικά προγράμματα:

Υπάρχουν δύο πλευρές κάθε λογικού προγράμματος: η δηλωτική (declarative) και η διαδικαστική (procedural). Η δηλωτική αφορά το περιεχόμενο ενός προγράμματος (τι μας λέει το πρόγραμμα), ενώ η διαδικαστική αφορά τον τρόπο με τον οποίο εξάγεται περιεχόμενο από το πρόγραμμα. Στην ιδανική περίπτωση, ο προγραμματιστής έχει ν' ασχοληθεί μόνο με τη δηλωτική πλευρά, να περιγράψει, δηλαδή, το πρόβλημα το οποίο θέλει να επιλύσει και ν' αφήσει τη διαδικαστική πλευρά (τον έλεγχο) στο σύστημα. Για παράδειγμα, ο κανόνας son(X, Y) :- father(Y, X), male(X). μπορεί να διαβαστεί με δύο τρόπους:

- Το *δηλωτικό* τρόπο: για κάθε X και Y, ο X είναι ο γιος του Y, αν ο Y είναι πατέρας του X και το X είναι αρσενικό.
- Το *διαδικαστικό* τρόπο: για ν' αποδείξουμε ότι 'ο X είναι γιος του Y', πρώτα αποδείξουμε ότι 'ο Y είναι πατέρας του X' κι έπειτα ότι 'το X είναι αρσενικό'.

Σημασία λογικών προγραμμάτων:

Ένα κοινό πρόβλημα σ' όλα τα προγράμματα, ανεξάρτητα από το εάν αντιμετωπίζονται με την προστακτική (imperative), την συναρτησιακή (functional) ή τη λογική προσέγγιση, είναι ο καθορισμός της ορθότητάς τους. Στο λογικό προγραμματισμό η ορθότητα ορίζεται σε σχέση με τη σημασία (meaning) και την επιθυμητή σημασία (intended meaning).

Η σημασία **Μ(Π)** ενός λογικού προγράμματος **Π**, ορίζεται ως το σύνολο των βασικών μοναδιαίων γεγονότων τα οποία είναι δυνατό να παραχθούν από το **Π** (deducible facts). Συνεπώς η σημασία ενός λογικού προγράμματος που αποτελείται μόνον από βασικά γεγονότα, είναι το ίδιο το πρόγραμμα. Αυτό σημαίνει ότι τα απλά προγράμματα, εννοούν ακριβώς αυτό που λένε. Από την άλλη μεριά, η επιθυμητή σημασία ενός λογικού προγράμματος **Π**, είναι το σύνολο *Μ* των βασικών γεγονότων, τα οποία ο χρήστης αναμένει να παραχθούν από το πρόγραμμα.

Λέμε ότι ένα πρόγραμμα Π είναι *σωστό* (*correct*) ως προς μια επιθυμητή σημασία *Μ*, αν η σημασία **Μ(Π)** του **Π** είναι υποσύνολο του *Μ*, αν δηλαδή, **Μ(Π)**$\subseteq$ **Μ**. Άρα ένα σωστό πρόγραμμα δεν λέει πράγματα τα οποία δεν είναι επιθυμητά.

Λέμε ότι ένα πρόγραμμα Π είναι *πλήρες* (*complete)* ως προς μια επιθυμητή σημασία *Μ*, αν το *Μ* είναι υποσύνολο του **Μ(Π)**, δηλαδή **Μ**$\subseteq$ **Μ(Π)**. Άρα ένα πλήρες πρόγραμμα, λέει οτιδήποτε είναι επιθυμητό. Σαν συμπέρασμα, ένα πρόγραμμα **Π** είναι ορθό και πλήρες ως προς μια επιθυμητή σημασία *Μ*, αν **Μ** = **Μ(Π)**.

Δομές Δεδομένων:

Οι δομές δεδομένων χρησιμοποιούνται για να οργανώσουν δεδομένα μ' έναν κατανοητό τρόπο και είναι σημαντικές στον προγραμματισμό γενικά, αλλά και στον προγραμματισμό ειδικότερα. Τα *δομημένα αντικείμενα* (*structured objects*) ή απλά *δομές* (*structures*), είναι αντικείμενα τα οποία έχουν αρκετά τμήματα (components) τα οποία, με τη σειρά τους, ενδέχεται να είναι δομές. Στο λογικό προγραμματισμό υπάρχουν δύο είδη δομών: οι *σύνθετοι όροι* (*compound terms*) και οι *λίστες* (*lists*). Η λίστα αποτελεί ειδική περίπτωση σύνθετου όρου.

Ένα από τα σημαντικότερα χαρακτηριστικά του λογικού προγραμματισμού είναι η ικανότητά του να χειρίζεται *λίστες όρων*. Οι λίστες υλοποιούνται χρησιμοποιώντας ένα ειδικό δυαδικό (τάξης 2) σύμβολο συνάρτησης, το '.', και μια ειδική σταθερά, τη 'nil', η οποία παριστάνει την *κενή λίστα*. Μια λίστα είναι είτε κενή (nil), είτε το πρώτο όρισμά της είναι ένα στοιχείο ενώ το δεύτερο μια λίστα. Έτσι, η λίστα τριών στοιχείων p, q, r, παριστάνεται από τον όρο .(a, .(b, .(c, nil))). Για να απλουστεύσουμε το συμβολισμό μας, χρησιμοποιούμε μια ειδική σύνταξη κατά την οποία ο όρος .(X, Y) γράφεται [X|Y]. Το X ονομάζεται *κεφαλή* (*head*) της λίστας και το Y *ουρά* (*tail*).

Αναιρέσιμη Λογική (Defeasible logic).

Η αναιρέσιμη λογική είναι μια απλή αλλά αποδοτική προσέγγιση στη μη μονότονη συλλογιστική. Αναπαριστά και διαχειρίζεται αντιφάσεις (conflicts) μεταξύ των κανόνων ενός προγράμματος. Οι αντιφάσεις εκφράζονται ως αντικρουόμενα συμπεράσματα. Μια μορφή αντίφασης είναι το συμπέρασμα ενός κανόνα να αποτελεί άρνηση του συμπεράσματος ενός άλλου κανόνα. Συνεπώς, η συλλογιστική αυτή χρησιμοποιεί την κλασική άρνηση της μαθηματικής λογικής (strong, classical negation) σε αντίθεση με την άρνηση ως αποτυχία που εφαρμόζουν οι γλώσσες λογικού προγραμματισμού, όπως για παράδειγμα η Prolog. Μια πιο πολύπλοκη αλλά χρήσιμη μορφή αντικρουόμενων συμπερασμάτων είναι όταν υπάρχουν πολλαπλά (θετικά) συμπεράσματα τα οποία εξουδετερώνουν το ένα το άλλο.

Η αναιρέσιμη λογική χαρακτηρίζεται ως σκεπτικιστική (skeptical) γιατί δεν επιτρέπει στους αντικρουόμενους κανόνες να ενεργοποιηθούν για να εξάγουν συμπεράσματα, συνεπώς διατηρείται συνεχώς η συνέπεια (consistency) των συμπερασμάτων. Για την διευθέτηση των αντιφάσεων μεταξύ των κανόνων μπορούν να χρησιμοποιηθούν προτεραιότητες μεταξύ των κανόνων.

Μια αναιρέσιμη θεωρία (ή μια αναιρέσιμη βάση γνώσης, ή αναιρέσιμο λογικό πρόγραμμα) περιλαμβάνει πέντε διαφορετικά είδη γνώσης : τα γεγονότα (facts), τους ισχυρούς κανόνες (strict rules), τους αναιρέσιμους κανόνες (defeasible rules), τους αναιρετές (defeaters) και τη σχέση υπεροχής.

Τα *γεγονότα* είναι αδιαπραγμάτευτες δηλώσεις, για παράδειγμα "Tweety is an emu" το οποίο εκφράζεται με τη μορφή *emu(tweety)*.

Οι *ισχυροί κανόνες* συμβολίζονται με A $\rightarrow$ p και είναι κανόνες όμοιοι με της κλασικής λογικής: "όταν η συνθήκη του κανόνα ισχύει οριστικά και αδιαμφισβήτητα τότε εξάγεται το συμπέρασμα". Παράδειγμα τέτοιου κανόνα είναι "Emus are birds" το οποίο γράφεται: *emu(X)*

→ *bird(X)*. Η εξαγωγή συμπερασμάτων αποκλειστικά με τη χρήση ισχυρών κανόνων ονομάζεται οριστική (definite inference).

Οι *αναιρέσιμοι κανόνες* συμβολίζονται A ⇒ p και είναι κανόνες που το συμπέρασμα τους μπορεί να αναιρεθεί από αντιφατικό συμπέρασμα. Ένα παράδειγμα είναι: "Birds typically fly" που γράφεται *bird(X) ⇒ flies(X)*. Η ιδέα είναι η εξής αν γνωρίζουμε ότι κάτι είναι πουλί μπορούμε να συμπεράνουμε ότι πετάει, εκτός κι αν υπάρχει άλλο, όχι κατώτερο, στοιχείο που ισχυρίζεται ότι ίσως δεν πετάει.

Οι *αναιρέτες* συμβολίζονται A ~> p και χρησιμεύουν για να αποτρέψουν ή αναιρέσουν την εξαγωγή κάποιου συμπεράσματος και όχι για την υποστήριξη συμπεράσματος. Ένα παράδειγμα τέτοιου κανόνα είναι: "Αν ένα ζώο είναι βαρύ τότε μπορεί να μην είναι ικανό να πετάει" και εκφράζεται *heavy(X) ~> ¬flies(X)*. Το σημαντικό είναι ότι η πληροφορία ότι ένα ζώο είναι βαρύ δεν είναι επαρκές στοιχείο για να καταλήξουμε ότι δεν πετάει. Είναι μόνο δεδομένο ότι το ζώο ίσως δεν μπορεί να πετάει. Με άλλα λόγια δεν επιδιώκει να μην συμπεράνουμε ότι δεν πετάει αν είναι βαρύ, απλά θέλει να εμποδίσει το συμπέρασμα ότι πετάει.

Η *σχέση υπεροχής* ανάμεσα στους κανόνες είναι μη κυκλική και χρησιμοποιείται για να προσδιορίσει προτεραιότητες μεταξύ των κανόνων, δηλαδή πότε ένας κανόνας υπερκαλύπτει το συμπέρασμα ενός άλλου κανόνα. Όταν ισχύει για δυο κανόνες η σχέση $r_1 > r_2$, τότε ο κανόνας r_1 ονομάζεται ανώτερος (superior) από τον r_2 και αντίστοιχα ο κανόνας r_2 κατώτερος (inferior) του r_1. Για παράδειγμα, έστω οι παρακάτω αναιρέσιμοι κανόνες των οποίων τα συμπεράσματα είναι αντικρουόμενα: r_1: bird(X) ⇒ flies(X) και r_2: brokenWing(X) ⇒ ¬flies(X) καμία απόφαση δεν μπορεί να παρθεί αν ένα πουλί με σπασμένα φτερά μπορεί να πετάξει ή όχι. Όμως αν εισάγουμε μια σχέση υπεροχής της μορφής $r_2 > r_1$ τότε μπορούμε να συμπεράνουμε ότι το πουλί δεν πετάει.

Ένα σημείο που πρέπει να τονίσουμε είναι ότι στην αναιρέσιμη λογική οι προτεραιότητες είναι τοπικές με την έννοια ότι δύο κανόνες θεωρούνται ότι είναι σε σύγκρουση αν έχουν συμπληρωματικές κεφαλές. Διαφορετικά μπορούν να συμμετέχουν σε σχέση υπεροχής η οποία δεν έχει καμία εφαρμογή στην θεωρία απόδειξης και εξαγωγής συμπερασμάτων.

Δεοντική Λογική.

Τροπική Λογική (Modal Logic):

Η δεοντική λογική είναι ένας κλάδος της φιλοσοφικής λογικής που αφορά συλλογισμούς με κανόνες. Έχει επηρεαστεί έντονα από την τροπική συλλογιστική (modal logic). Ένα modal είναι μια έκφραση (όπως 'απαραίτητα' ή πιθανά) που χρησιμοποιείται για να πιστοποιήσει την αλήθεια ενός ισχυρισμού. Η τροπική λογική είναι η μελέτη της συμπερασματικής συμπεριφοράς των εκφράσεων "είναι αναγκαίο να" που συμβολίζεται ως □ και "είναι πιθανό να" που συμβολίζεται ως ◊.

Μια διαδεδομένη τροπική λογική στηρίχτηκε σε μια αδύναμη σχετικά λογική που ονομάζεται **K** (από τον Saul Kripke). Τα σύμβολα της **K** περιλαμβάνουν το '~' για το 'not', '→' για το 'εάν…τότε', and '□' για τον τροπικό τελεστή 'είναι απαραίτητο να'. Τα συνδετικά '&', '∨', και '↔' μπορούν να οριστούν από τα '~' και '→'.

Η **K** λογική οδήγησε στην προσθήκη του παρακάτω κανόνα στις αρχές της προτασιακής λογικής:

Κανόνας αναγκαιότητας: Εάν A είναι ένα θεώρημα της **K** τότε είναι και το $\Box A$.

Κατανεμημένο Αξίωμα: $\Box(A \rightarrow B) \rightarrow (\Box A \rightarrow \Box B)$.

Σύμφωνα με τον κανόνα αναγκαιότητας κάθε θεώρημα της λογικής είναι υποχρεωτικό και το αξίωμα υποστηρίζει ότι αν είναι αναγκαίο το 'αν A τότε B' τότε αν είναι υποχρεωτικό το A είναι υποχρεωτικό και το B να ισχύει.

Ο τελεστής $\Diamond$ (για το πιθανό) μπορεί να οριστεί από το $\Box$ ως $\Diamond A = \sim\Box\sim A$.

Δεοντική Λογική (Deontic Logic):

Οι δεοντικές λογικές εισάγει το σύμβολο O για την έκφραση 'είναι υποχρεωτικό να', από το οποίο τα σύμβολα P για την έκφραση 'είναι επιτρεπτό να' και F για την έκφραση 'είναι απαγορευτικό να' ορίζονται ως εξής: $PA = \sim O\sim A$ and $FA = O\sim A$. Ένα βασικό σύστημα **D** δεοντικής λογικής μπορεί να θεμελιωθεί προσθέτοντας το παρακάτω αξίωμα (D) στην **K**:

$$(D)\ \ OA \rightarrow PA$$

Το αξίωμα (D) εγγυάται την συνέπεια του συστήματος υποχρεώσεων εισάγοντας το εξής 'όταν το A είναι υποχρεωτικό, είναι και επιτρεπτό'. Ένα σύστημα το οποίο μας υποχρεώνει να ισχύει το A αλλά δεν μας επιτρέπει να το κάνουμε αυτό οδηγεί σε αναπόφευκτους δεσμούς. Αν και μερικοί επιστήμονες θεωρούν ότι τέτοιες συγκρούσεις είναι εφικτές οι περισσότεροι αποδέχονται την ισχύ του (D).

Το αξίωμα $O(OA \rightarrow A)$ είναι ένα ακόμα αξίωμα που είναι επιθυμητό. Αν και είναι λάθος να θεωρούμε ότι εάν το A είναι υποχρεωτικό τότε ισχύει το A πολλοί ερευνητές της δεοντικής λογικής θεωρούν ότι το **D** πρέπει να συμπεριλάβει και αυτό το αξίωμα.

Μια άλλη δεοντική λογική είναι η D^* η οποία περιλαμβάνει τον κανόνα

$$\frac{P \rightarrow Q}{OP \rightarrow OQ}$$

και τρία αξιώματα

- $(OP \wedge OQ) \Leftrightarrow O(P \wedge Q)$
- Oτ (αυτό που πρέπει να είναι είναι υποχρεωτικό)
- O$\perp$ (τίποτα μη δυνατό είναι υποχρεωτικό)

Ενώ η D^* έχει κάποιες επιθυμητές ιδιότητες δεν χρησιμοποιούνται τόσο λόγω την ύπαρξη πολλών παραδόξων.

Standard Δεοντική Λογική (SDL):

Ο Von Wright πρότεινε την δεοντική λογική που περιλαμβάνει τα εξής αξιώματα και κανόνες (οι κανόνες και οι παραγόμενοι κανόνες είναι της μορφής $\varphi_1, \ldots, \varphi_\nu / \psi$):

0.Όλες οι ταυτολογίες του προτασιακού λογισμού.

1. $Op \equiv \neg P \neg p$
2. $Pp \vee P \neg p$
3. $P(p \vee q) \equiv Pp \vee Pq$
4. $p \equiv q \,/\, Pp \equiv Pq$

Στην συνέχεια διαπιστώθηκε ότι αυτό το σύστημα ήταν πολύ κοντά στην κανονική τροπική λογική και ενδυνάμωσε τη σημασιολογία του Kripke με χρήση ως βασικό τελεστή το τον τελεστή αναγκαιότητας O και την αποδοχή της εγκυρότητας του:

(Oτ) $O(p \vee \neg p)$ που ο Von Wright δεν δεχόταν.

Το σύστημα του Von Wright ως σύστημα κανονικής τροπικής λογικής περιλαμβάνει τα εξής αξιώματα και κανόνες:

0. Όλες οι ταυτολογίες του προτασιακού λογισμού.
1. $O(p \supset q) \supset (Op \supset Oq)$ το αξίωμα K
2. $Op \supset Pp$
3. $Pp \supset \neg O \neg p$ η άδεια είναι το δυικό της υποχρέωσης
4. $Fp \supset \neg Pp$ ότι είναι απαγορευμένο δεν είναι και επιτρεπτό.

5. Νόμος του θέτει: $p, p \supset p / p$
6. Ο-αναγκαιότητα: p / Op

Η λογική αυτή ονομάζεται *Standard Δεοντική Λογική*.

Βιβλιογραφία

Books:

- Τεχνητή Νοημοσύνη
  Β' Έκδοση (Γαρταγάνης)
  Βλαχάβας Ιωάννης, Κεφαλάς Πέτρος, Βασιλειάδης Νικόλαος,
  Κόκκορας Φώτης, Σακελλαρίου Ηλίας (2005) Θεσσαλονίκη
- Deontic Logic in Computer Science
  Normative System Specification
  Edited by John-Jules Ch. Meyer and Roel J. Wieringa
  Wiley Professional Computing
  Chi Chester-New York-Brisbane-Toronto-Singapore

Papers:

- A logic for Default Reasoning
  R. Reiter (1980)
  Department of Computer Science, University of British
  Columbia, Vancouver, B.C., Canada
  Recommended by Patrick J. Hayes
  Copyright by North Holland Publishing Company
- A tutorial on Default Logics
  Grigoris Antoniou (1999)
  Griffith University
  ACM Computing Surveys, Vol. 31, No. 3, September 1999
- Operational Concepts of Nonmonotonic Logics
  Part 1: Default Logic
  Grigoris Antoniou and V. Sperschneider (1994)
  University of Osnabruck, Department of Mathematics and Computer Science,
  Albrechtstrasse 28, D-4500 Osnabruck, Germany
- Two Approaches to the Formalization of Defeasible Deontic Reasoning
  Henry Prakken (1996)
  Computer / Law Institute
  Faculty of Law, Free University
  De Boelelaan 1105, 1081 Amsterdam
  The Neverlands
  April 26, 1996
- Defeasible Logic
  Donald Nute (2003)
  Department of Philosophy and Artificial Intelligence Center,
  The University of Georgia, Athens, GA 30605, U.S.A.,
  dnute@uga.edu
  O. Bartenstein et al. (Eds.): INAP 2001, LNAI 2543, pp. 151–169, 2003.
  Copyright Springer-Verlag Berlin Heidelberg 2003

Sites:

- *http://mally.stanford.edu/deontic.html*
- *http://plato.stanford.edu/entries/logic-ai/*
- *http://www.w3schools.com/*
- *http://lpis.csd.auth.gr/prolog/Logic_Programming/node3.html#SECTION00021000000000000000*

Εικόνα Εξωφύλλου:

- *http://gauntlet.ucalgary.ca/story/10343*
  Artificial intelligence at the U of C
  Splicing neurons and micro-chips blurs the line between sci-fi and reality
  Johanna Hung