Εθνικό Μετσόβιο Πολυτεχνείο
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ &
ΥΠΟΛΟΓΙΣΤΩΝ

Βελτιστοποιημένοι Αλγόριθμοι Επαναγραφής Ερωτημάτων για Εκφραστικές Περιγραφικές Λογικές

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

της

Δέσποινας Τριβέλα

Διπλωματούχου Ηλεκτρολόγου Μηχανικού &
Μηχανικού Υπολογιστών Ε.Μ.Π. (2009)

Αθήνα, Δεκέμβριος 2015

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
& ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ & ΥΠΟΛΟΓΙΣΤΩΝ

Βελτιστοποιημένοι Αλγόριθμοι Επαναγραφής Ερωτημάτων για Εκφραστικές Περιγραφικές Λογικές

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

της

Δέσποινας Τριβέλα

Διπλωματούχου Ηλεκτρολόγου Μηχανικού &
Μηχανικού Υπολογιστών Ε.Μ.Π. (2009)

Συμβουλευτική Επιτροπή: Γιώργος Στάμου
Ανδρέας Γεώργιος Σταφυλοπάτης
Στέφανος Κόλλιας

Εγκρίθηκε από την επταμελή εξεταστική επιτροπή την 18η Δεκεμβρίου 2015.

... Γ. Στάμου Επίκουρος Καθηγητής Ε.Μ.Π.	... Α.Γ. Σταφυλοπάτης Καθηγητής Ε.Μ.Π.	... Σ. Κόλλιας Καθηγητής Ε.Μ.Π.
... Π. Τσανάκας Καθηγητής Ε.Μ.Π.	... Μ. Κουμπαράκης Καθηγητής Ε.Κ.Π.Α.	... Δ. Φωτάκης Αναπληρωτής Καθηγητής Ε.Μ.Π.
	... Β. Βασσάλος Αναπληρωτής Καθηγητής Ο.Π.Α.	

Αθήνα, Δεκέμβριος 2015

...

Δέσποινα Τριβέλα
Διδάκτωρ Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Στον Βασίλη, τον Χάρη ...

«Θεωρώ πιο γενναίο εκείνον που κυριαρχεί στα πάθη του από εκείνον που κυριαρχεί στους εχθρούς του. Η δυσκολότερη νίκη είναι εκείνη ενάντια στον ίδιο σου τον εαυτό»

Αριστοτέλης

Ευχαριστίες

Ολοκληρώνοντας τη διδακτορική μου διατριβή θα ήθελα να εκφράσω τις ευχαριστίες μου στον επίκουρο καθηγητή Γ. Στάμου, επιβλέποντα της παρούσας εργασίας, για τη δυνατότητα που μου έδωσε να εργασθώ στην έρευνα, τη καθοδήγησή του, και κυρίως για τη στήριξη, πρακτική και ηθική, που μου προσέφερε αυτά τα χρόνια. Επίσης, θα ήθελα να ευχαριστήσω τον καθηγητή Σ. Κόλλια, ο οποίος με καλοσώρισε στην ομάδα του εργαστηρίου παρέχοντας την υποστήριξή του όποτε χρειάστηκε.

Αισθάνομαι την ανάγκη να ευχαριστήσω θερμά δύο ξεχωριστούς ανθρώπους με τους οποίους είχα την τύχη να συνεργαστώ, τον Αλέξανδρο Χορταρά και τον Γιώργο Στοΐλο. Θα ήθελα να ευχαριστήσω τον Αλέξανδρο, με τον οποίο συνεργάζομαι από το ξεκίνημα της διατριβής μου, για την καθοριστική συμβολή του στην εκπόνηση αυτής της εργασίας. Αισθάνομαι ευγνώμων για τη βοήθεια, τη συνέπεια και τη διακριτικότητα του χαρακτήρα του. Θα ήθελα να ευχαριστήσω τον Γιώργο για την πολύτιμη συνεισφορά του. Τον ευχαριστώ θερμά για τη διαρκή του καθοδήγηση, τις παρατηρήσεις, την αστείρευτη διάθεσή του για έρευνα και τις συμβουλές του με ερευνητικό και εκπαιδευτικό χαρακτήρα.

Θα ήθελα να εκφράσω ευχαριστίες σε όλους όσοι υπήρξαν μέλη του εργαστηρίου κατά τη διάρκεια της διδακτορικής μου διατριβής, για το κλίμα συναδελφικότητας και τις ευχάριστες στιγμές. Ιδιαίτερα, ευχαριστώ τους φίλους και συναδέλφους Γιάννη Καλαντίδη, Τάσο Βενέτη και Γιώργο Τόλια για την υποστήριξη, άλλοτε ψυχολογική και άλλοτε πρακτική. Ευχαριστώ τις φίλες Άννα Γκατζιούρα και Πένυ Καρανάσου για την εμπιστοσύνη τους.

Δέσποινα Τριβέλα
Αθήνα, Δεκέμβριος 2015

ΠΕΡΙΛΗΨΗ

Μία σημαντική προσέγγιση για την απάντηση ερωτημάτων σε οντολογίες είναι η *επαναγραφή* της οντολογίας και του ερωτήματος σε φορμαλισμούς για τους οποίους έχουν αναπτυχθεί αλγόριθμοι απάντησης ερωτημάτων. Μία βασική οικογένεια αλγορίθμων επαναγραφής περιλαμβάνει εκείνους που βασίζονται στην *επίλυση*. Χαρακτηριστικό τους είναι η δυνατότητα γενίκευσης σε διαφορετικές γλώσσες οντολογίας, καθώς και η δυνατότητα αξιοποίησης των τεχνικών βελτιστοποίησης που έχουν προκύψει από την εκτενή και μακροχρόνια έρευνα στον κλάδο της θεωρίας αποδείξεων με βάση την επίλυση. Την ίδια στιγμή όμως, η ικανότητα γενίκευσης που προσφέρει η μέθοδος της επίλυσης, μπορεί έχει επιπτώσεις στην επίδοση των αλγορίθμων.

Στην παρούσα διατριβή μελετάμε και βελτιστοποιούμε υπάρχουσες μεθόδους που εφαρμόζουν τεχνικές επίλυσης με σκοπό να σχεδιάσουμε αποδοτικούς αλγορίθμους επαναγραφής για σημαντικά υποσύνολα της OWL 2. Η βασική ιδέα των αλγορίθμων είναι να αναπτύσσουν μια στρατηγική εφαρμογής κανόνων που θα λαμβάνει υπόψη τα ιδιαίτερα χαρακτηριστικά της κάθε γλώσσας περιγραφικής λογικής. Αρχικά, παρουσιάζουμε έναν αλγόριθμο για τη γλώσσα DL-Lite που σχετίζεται άμεσα με την OWL 2 QL. Ο λογισμός μας αποφεύγει την εκτέλεση περιττών κανόνων συμπερασμού επιλύοντας έτσι, ένα από τα βασικά προβλήματα αντίστοιχων αλγορίθμων. Στη συνέχεια, επεκτείνουμε τον αλγόριθμο για τη γλώσσα $\mathcal{ELHI}$ που σχετίζεται με την OWL 2 EL. Παρά την υψηλότερη εκφραστικότητα της γλώσσας, δείχνουμε ότι απαιτούνται λίγες μόνο τροποποιήσεις στον αλγόριθμο για την DL-Lite. Τέλος, επεκτείνουμε περαιτέρω τον αλγόριθμο, ώστε να υποστηρίζει την υψηλής εκφραστικότητας γλώσσα Horn-$\mathcal{SHIQ}$. Ακολουθώντας την ίδια λογική ο αλγόριθμος λαμβάνει υπόψη τις σύνθετες, αλλά συγκεκριμένες αλληλεπιδράσεις των αξιωμάτων της γλώσσας, ώστε να περιορίσει στο ελάχιστο την εφαρμογή των κανόνων της επίλυσης.

Υλοποιήσαμε τους αλγορίθμους σε ένα πρωτότυπο σύστημα και διεξάγαμε πειραματική αξιολόγηση κάνοντας χρήση σύνθετων OWL οντολογιών μεγάλης κλίμακας. Πρόκειται για την πρώτη αξιολόγηση συστήματος επαναγραφής με οντολογίες τέτοιου μεγέθους. Στην περίπτωση των οντολογιών σε DL-Lite και $\mathcal{ELHI}$ το σύστημά μας αποκρίνεται ταχύτερα από ότι τα διαθέσιμα συστήματα. Στις περιπτώσεις οντολογιών σε Horn-$\mathcal{SHIQ}$ το σύστημά μας αποκρίνεται σε μερικά δευτερόλεπτα όταν το μοναδικό υπάρχον σύστημα που υποστηρίζει αυτή την εκφραστικότητα δεν καταφέρνει να αποκριθεί.

Λέξεις Κλειδιά

Περιγραφικές Λογικές, Αλγόριθμοι Επαναγραφής Ερωτημάτων, Επίλυση, Οντολογία.

ABSTRACT

An important approach to query answering over OWL ontologies is via *rewriting* the input ontology and query into a new set of axioms that are expressed in logics for which scalable query answering algorithms exist. An important family of rewriting algorithms are the *resolution-based* ones, mostly due to their ability to adapt to any ontology language and the long years of research in resolution theorem-proving. However, this generality comes with performance prices.

In the current work we revisit and refine the resolution approaches in order to design efficient rewriting algorithms for many important fragments of OWL 2. The underlying idea of our algorithms is to apply a resolution-based strategy that takes into account the specific structure of the axioms supported by each language. First, we present an algorithm for the language DL-Lite, which is strongly related to OWL 2 QL. Our calculus is restricted so that it avoids performing many unnecessary inferences, one of the main problems of these algorithms. Subsequently, we extend our algorithm to the language $\mathcal{ELHI}$ which is strongly related to OWL 2 EL. This is a difficult task as $\mathcal{ELHI}$ is an expressive language, however, we show that the calculus for DL-Lite requires relatively small extensions. Finally, we extend our calculus for the very expressive language Horn-$\mathcal{SHIQ}$. By following the same approach as in the case of $\mathcal{ELHI}$, our calculus takes into account the complex axioms interactions in order to restict the resolution rules application.

We have implemented our algorithms and we have conducted an extensive experimental evaluation using many well-known large and complex OWL ontologies. This is the first evaluation of rewriting algorithms of this magnitude. In the case of DL-Lite and $\mathcal{ELHI}$ ontologies our results show that our system is in many cases several orders of a magnitude faster than several sofisticated rewriting systems. Similarly, in the case of Horn-$\mathcal{SHIQ}$ ontologies, our system greatly outperforms the only available rewriting system for Horn-$\mathcal{SHIQ}$.

Keywords

Description Logics, Query Rewriting Algorithms, Resolution, Ontology.

Περιεχόμενα

Κατάλογος Πινάκων

Κατάλογος Αλγορίθμων

Κατάλογος Συντμήσεων

ανν	:	αν και μόνο αν
μκε	:	μέγιστος κοινός ενοποιητής
ΠΛ	:	Περιγραφικές Λογικές
ΣΕ	:	Συζευκτικό Ερώτημα
ΣΣΕ	:	Σύνολο Συζευκτικών Ερωτημάτων
ABox	:	Assertional Box
$\mathcal{ALC}$	:	$\mathcal{A}$ttributive $\mathcal{L}$anguage with $\mathcal{C}$omplement: $\top \mid \bot \mid A \mid \neg C \mid \exists R.C \mid \forall R.C$
DL	:	Description Logics
$\mathcal{EL}$	:	Η γλώσσα Περιγραφικών Λογικών $\mathcal{EL}$
$\mathcal{ELHI}$	:	Η γλώσσα Περιγραφικών Λογικών $\mathcal{EL}$ με αξιώματα υπαγωγής ρόλων $\mathcal{H} : R \sqsubseteq S$ και αντίστροφους ρόλους $(\mathcal{I}) : R^-$
OWL	:	Web Ontology Language
$\mathcal{S}$	:	Η γλώσσα Περιγραφικών Λογικών $\mathcal{ALC}$ με αξιώματα μεταβατικών ρόλων $(\mathcal{ALC}_{R^+}) : \mathsf{Tr}(R)$
$\mathcal{SI}$	:	Η γλώσσα Περιγραφικών Λογικών $\mathcal{S}$ με αντίστροφους ρόλους $(\mathcal{I}) : R^-$
$\mathcal{SHI}$	:	Η γλώσσα Περιγραφικών Λογικών $\mathcal{SI}$ με αξιώματα υπαγωγής ρόλων $(\mathcal{H}) : R \sqsubseteq S$
$\mathcal{SHIQ}$	:	Η γλώσσα Περιγραφικών Λογικών $\mathcal{SHI}$ με προσοντούχο περιορισμό πληθυκότητας $(\mathcal{Q}) :\leq nR, \geq nR$
Horn-$\mathcal{SHIQ}$	:	Η γλώσσα Περιγραφικών Λογικών $\mathcal{SHIQ}$ που δεν επιτρέπει στο δεξί σκέλος των αξιωμάτων υπαγωγής τις έννοιες $C_1 \sqcup C_2, \leq mR.C, m > 1$ και στο αριστερό τις έννοιες $\neg C, \forall R.C, \geq nR.C, n > 1, \leq mR.C$
TBox	:	Terminological Box

Μέρος I

Θεμέλια

Κεφάλαιο 1

Εισαγωγή

Η *αναπαράσταση γνώσης* (*knowledge representation*) αποτελεί σημαντικό κλάδο της Τεχνητής Νοημοσύνης (Artificial Intelligence), που αφορά τον τρόπο αναπαράστασης και καταγραφής της ανθρώπινης γνώσης σε κάποιο υπολογιστικό σύστημα. Η γνώση αναπαρίσταται αντιστοιχίζοντας αντικείμενα και έννοιες του πραγματικού κόσμου σε σύμβολα. Η χρήση των συμβόλων επιτρέπει την κωδικοποίηση της γνώσης και την επεξεργασία της με μαθηματικό τρόπο, δηλαδή με τυπικούς κανόνες από τους οποίους εξάγονται ορθά συμπεράσματα.

Σκοπός της αναπαράστασης γνώσης είναι η ανάπτυξη *συστημάτων γνώσης* (*knowledge based systems*). Πρόκειται για συστήματα που βασίζουν τη λειτουργία τους στη συμβολική αναπαράσταση γνώσης που έχουν στη διάθέση τους, η οποία ονομάζεται *βάση γνώσης* (*knowledge base*). Τέτοιου είδους συστήματα αξιοποιούν τις διαθέσιμες περιγραφές προκειμένου να καταλήξουν σε λογικά συμπεράσματα. Έτσι, η απόκρισή τους δεν σχετίζεται με απλή ανάκτηση πληροφορίας, αλλά με λογικές συνεπαγωγές από τη βάση γνώσης. Οι αυτοματοποιημένες διαδικασίες που εφαρμόζουν κατά την εξαγωγή των συμπερασμάτων, περιγράφονται από τους μηχανισμούς *συλλογιστικής* (*reasoning*).

Η *λογική* (*logic*) είναι η επιστήμη που μελετά τη συλλογιστική και ερευνά τους μηχανισμούς με τους οποίους μπορούν να προκύψουν συμπεράσματα από τη δεδομένη γνώση. Η επιστήμη της λογικής αφορά τόσο τον κλάδο της φιλοσοφίας όσο και τους κλάδους των μαθηματικών και της επιστήμης των υπολογιστών. Η πρώτη προσέγγιση αναπαράστασης της γνώσης με τυπικό τρόπο έγινε από τον Αριστοτέλη που έθεσε τα θεμέλια της *τυπικής λογικής* (*formal logic*) [74, 119].

Μια ιδιαίτερα σημαντική γλώσσα αναπαράστασης γνώσης είναι η *Λογική Πρώτης Τάξης, ΛΠΤ* (*First Order Logic, FOL*) [93, 134, 15]. Τα σύμβολα που χρησιμοποιεί μια γλώσσα αναπαράστασης γνώσης ορίζουν το *αλφάβητό* (*alphabet*) της, ενώ οι κανόνες βάσει των οποίων τα σύμβολα συνδυάζονται μεταξύ τους ορίζουν το *συντακτικό* (*syntax*) της γλώσσας. Στη λογική πρώτης τάξης τα σύμβολα διακρίνονται σε λογικά, που είναι οι μεταβλητές $x, y, \ldots$, οι τελεστές $\neg, \vee, \leftarrow, \approx, \ldots$, οι ποσοδείκτες $\exists, \forall$, και σε μη-λογικά που είναι τα σύμβολα συναρτήσεων $f, g, \ldots$, και τα κατηγορήματα, $P, R, \textsf{Φοιτητής}, \ldots$. Για παράδειγμα, η παρακάτω είναι μία πρόταση λογικής πρώτης τάξης:

$$\forall x.(\textsf{Φοιτητής}(x) \leftarrow \textsf{ΜεταπτυχιακόςΦοιτητής}(x))$$

όπου τα Φοιτητής, ΜεταπτυχιακόςΦοιτητής είναι κατηγορήματα, ο κατασκευαστής $\forall$ ονομάζεται καθολικός ποσοδείκτης και η μεταβλητή x που τον ακολουθεί, καθολική μεταβλητή. Οι κανόνες που εφαρμόζονται στα σύμβολα κατά τη διαδικασία συλλογιστικής, καταλήγουν σε αναντίρρητα συμπεράσματα που δεν εξαρτώνται από τη σημασία-ερμηνεία της πληροφορίας που περιγράφεται. Οι ερμηνείες αποδίδονται στους επιτρεπτούς συνδυασμούς συμβόλων από τη *σημασιολογία* (*semantics*) της γλώσσας. Με άλλα λόγια, η σημασιολογία προσδιορίζει τη σχέση ανάμεσα στα σύμβολα και τα αντικείμενα του κόσμου που περιγράφουν. Στη λογική πρώτης τάξης η σημασιολογία προσδιορίζεται με τη χρήση μοντέλων. Ένα *μοντέλο* (*model*) αποτελείται από ένα σύνολο με στοιχεία που αναπαριστούν αντικείμενα του πραγματικού κόσμου. Έτσι, για παράδειγμα, η πρόταση $\forall x.$Φοιτητής(x) είναι αληθής όταν η μεταβλητή x αντιστοιχίζεται σε στοιχεία του μοντέλου που αναπαριστούν φοιτητές. Επομένως, η πρόταση που διατυπώσαμε παραπάνω δηλώνει ότι εαν κάποιος είναι μεταπτυχιακός φοιτητής, τότε είναι φοιτητής. Σαν ένα ακόμα παράδειγμα πρότασης λογικής πρώτης τάξης, παραθέτουμε την παρακάτω:

$$\forall x.(\exists y.\text{έχειΠτυχίο}(x, y) \leftarrow \text{ΜεταπτυχιακόςΦοιτητής}(x))$$

όπου το έχειΠτυχίο είναι ένα δυαδικό κατηγόρημα, ο κατασκευαστής $\exists$ ονομάζεται υπαρξιακός τελεστής, και η μεταβλητή y που τον συνοδεύει, υπαρξιακή. Η πρόταση δηλώνει ότι οι μεταπτυχιακοί φοιτητές έχουν κάποιο πτυχίο. Τέλος, σαν παράδειγμα συλλογιστικής θεωρούμε τη βάση γνώσης που περιλαμβάνει την παραπάνω πρόταση καθώς και τις $\forall x.(\text{Πτυχιούχος}(x) \leftarrow \text{έχειΠτυχίο}(x, y))$ και ΜεταπτυχιακόςΦοιτητής(Άννα) Συμπεραίνουμε την πρόταση Πτυχιούχος(Άννα), ότι δηλαδή η Άννα είναι πτυχιούχος.

1.1 Περιγραφικές Λογικές

Οι *Περιγραφικές Λογικές, ΠΛ* (*Description Logics, DL*) [7] είναι μια οικογένεια γλωσσών αναπαράστασης γνώσης που αποτελούν ένα *εκφραστικό* και *εύρωστα αποφάνσιμο* (*robustly decidable*) υποσύνολο της λογικής πρώτης τάξης [25]. Οι περιγραφικές λογικές χαρακτηρίζονται από τυπική σημασιολογία και απλότητα στην κατανόηση, με αποτέλεσμα να παρέχουν τη θεωρητική βάση για πολλά συστήματα αναπαράστασης γνώσης.

Η έρευνα στην περιοχή των περιγραφικών λογικών προέκυψε από τις ανάγκες των συστημάτων γνώσης. Οι γλώσσες περιγραφικής λογικής διαθέτουν εκφραστικές δυνατότητες που ανταποκρίνονται στις ανάγκες συστημάτων με διαφορετικά πεδία εφαρμογής. Την ίδια στιγμή, επιτρέπουν τη χρήση μηχανισμών συλλογιστικής που έχουν μελετηθεί εκτενώς ως προς την υπολογιστική τους ικανότητα. Η εκφραστική δύναμη των περιγραφικών λογικών σε συνδυασμό με την πολυπλοκότητα και την ικανότητα απόκρισης των μηχανισμών συλλογιστικής που επιτρέπουν, είχε ως συνέπεια την ευρεία χρήση τους κατά το σχεδιασμό των συστημάτων γνώσης. Σημειώνουμε ότι οι περιγραφικές λογικές αποτελούν το θεωρητικό υπόβαθρο για την γλώσσα ανάπτυξης οντολογιών OWL (Web Ontology Language) [16, 107, 97, 63] που αναπτύχθηκε για τις ανάγκες του Σημασιολογικού Ιστού (Semantic Web - SW) [20, 120]. Ο σημασιολογικός ιστός αποτελεί μία επέκταση του *Παγκόσμιου Ιστού* (*World Wide Web -*

WWW) [18, 17, 19], όπου η πληροφορία αποκτά πλέον σημασιολογικό περιεχόμενο, το οποίο γίνεται κατανοητό και αξιοποιείται με αυτόματο τρόπο από τις μηχανές. Κατά τη διάρκεια των χρόνων έχει προταθεί μια πληθώρα επεκτάσεων περιγραφικών λογικών όπως είναι οι Μη-Μονότονες Λογικές (NonMonotonic Logics) [49], οι Χρονικές Λογικές (Temporal Logics) [112], Ασαφής Λογική (Fuzzy Logic) [80], και άλλες.

Οι *μηχανισμοί συλλογιστικής* (*reasoning algorithms*) των γλωσσών περιγραφικής λογικής αντιμετωπίζουν βασικά προβλήματα, όπως είναι ο έλεγχος ικανοποιησιμότητας, ο έλεγχος υπαγωγής και η απάντηση ερωτημάτων. Ένας σημαντικός μηχανισμός συλλογιστικής που χρησιμοποιείται στις περιγραφικές λογικές, και γενικότερα στη λογική πρώτης τάξης, που θα μας απασχολήσει σε αυτή την εργασία, είναι η τεχνική της *επίλυσης* (*resolution*) [72]. Επίσης, μία άλλη σημαντική τεχνική είναι η μέθοδος *πινάκων* (*tableaux*) [118]. Έχουν αναπτυχθεί αρκετά συστήματα που υλοποιούν αλγορίθμους συλλογιστικής για περιγραφικές λογικές, ονομάζονται *μηχανές συλλογιστικής* (*reasoners*) [46, 124, 59, 100], και όπως έχει αποδειχθεί, συμπεριφέρονται πολύ καλά στην πράξη.

Οι Περιγραφικές Λογικές (ΠΛ) όπως υποδεικνύει το όνομά τους περιγράφουν τη γνώση ενός πεδίου ενδιαφέροντος με έννοιες, ενώ η σημασιολογία τους βασίζεται στην τυπική λογική. Το αλφάβητο των περιγραφικών λογικών αποτελείται από τα *άτομα* (*individuals*), τις *ατομικές έννοιες* (*atomic concepts*) και τους *ατομικούς ρόλους* (*atomic roles*). Επιπλέον, το αλφάβητό τους περιλαμβάνει *κατασκευαστές* (*constructors*), με τους οποίους οι ατομικές έννοιες συνδυάζονται για να περιγράψουν τις *σύνθετες έννοιες* (*complex concepts*). Μια από τις πιο βασικές γλώσσες ΠΛ είναι η $\mathcal{ALC}$ [117] που υποστηρίζει το σύνολο των κατασκευαστών $\{\neg, \sqcap, \sqcup, \forall, \exists\}$. Η σύνταξη της $\mathcal{ALC}$ ορίζεται ως εξής:

$$A \mid \top \mid \bot \mid \neg C \mid C \sqcap D \mid C \sqcup D \mid \forall R.C \mid \exists R.\top$$

όπου A είναι μία ατομική έννοια, R ένας ατομικός ρόλος, C, D σύνθετες έννοιες, $\top$ είναι η *καθολική έννοια* (*universal concept*), και $\bot$ είναι η *κενή έννοια* (*bottom concept*). Έτσι για παράδειγμα, αν θεωρήσουμε την ατομική έννοια Φοιτητής, και τον ρόλο έχειΠτυχίο μπορούμε να περιγράψουμε τον μεταπτυχιακό φοιτητή με τη σύνθετη έννοια:

$$\text{Φοιτητής} \sqcap \exists \text{έχειΠτυχίο}$$

Οι ΠΛ πέρα από την περιγραφή σύνθετων εννοιών δίνουν τη δυνατότητα καθορισμού σχέσεων μεταξύ αυτών. Για παράδειγμα μπορούμε να περιγράψουμε ότι κάθε μεταπτυχιακός φοιτητής είναι φοιτητής με ένα *αξίωμα υπαγωγής εννοιών* (*concept inclusion*):

$$\text{Μεταπτυχιακός} \sqsubseteq \text{Φοιτητής}$$

Επιπλέον, ορίζουμε τη σύνθετη έννοια του μεταπτυχιακού φοιτητή με ένα *αξίωμα ισοδυναμίας* (*equivalent concepts axiom*)

$$\text{Μεταπτυχιακός} \equiv \text{Φοιτητής} \sqcap \exists \text{έχειΠτυχίο}$$

Οι κατασκευαστές που υποστηρίζει η γλώσσα περιγραφικής λογικής καθορίζουν την εκφραστικότητά της. Για παράδειγμα, μία γλώσσα περιγραφικής λογικής που περιλαμβάνει τον *προσοντούχο περιορισμό πληθυκότητας* (*qualified number restriction*),

επιτρέπει να ορίσουμε τον επί πτυχίω φοιτητή, δηλαδή τον φοιτητή που του απομένουν το πολύ 3 μαθήματα, ως εξής:

$$\text{Φοιτητής} \sqcap \leq 3\text{οφείλει.Μάθημα}$$

Εαν επεκτείνουμε την $\mathcal{ALC}$ ώστε να υποστηρίζει *μεταβατικούς ρόλους* (*transitive roles*) προκύπτει η γλώσσα $\mathcal{S}$. Για παράδειγμα, η γλώσσα $\mathcal{S}$ επιτρέπει να δηλώσουμε ότι ο ρόλος έχειΠροαπαιτούμενοΜάθημα είναι μεταβατικός. Έτσι, εαν γνωρίζουμε ότι το μάθημα Β προαπαιτεί το Α και το Γ το Β, τότε συμπεραίνουμε ότι το Γ προαπαιτεί και το Α. Η γλώσσα $\mathcal{SH}$ επιτρέπει να ορίζουμε *ιεραρχίες ρόλων* (*role hierarchies*), $\mathcal{H}$, για παράδειγμα, έχειΠρόγραμμαΜεταπτυχιακώνΣπουδών $\sqsubseteq$ έχειΠρόγραμμαΣπουδών. Η γλώσσα $\mathcal{SHI}$ [64] προκύπτει από τη $\mathcal{SH}$, αν επεκταθεί με *αντίστροφους ρόλους* (*role inverses*), $\mathcal{I}$. Έτσι, για παράδειγμα, ο ρόλος διδάσκειΜάθημα που συσχετίζει κάποιον καθηγητή με ένα μάθημα, έχει αντίστροφο τον διδάσκεταιΑπό, που συσχετίζει ένα μάθημα με κάποιο καθηγητή. Επιπλέον, η $\mathcal{SHIN}$ υποστηρίζει τον *περιορισμό πληθυκότητας* (*number restriction*), $\mathcal{N}$, και η $\mathcal{SHIQ}$ υποστηρίζει τον προσοντούχο περιορισμό πληθυκότητας, $\mathcal{Q}$. Εάν η γλώσσα ΠΛ επιτρέπει *ονοματικές έννοιες* (*nominals*), $\mathcal{O}$, τότε προκύπτουν οι εκφραστικές $\mathcal{SHOIN}, \mathcal{SHOIQ}$ [65, 66] καθώς και $\mathcal{SROIQ}$ [62] που επιτρέπει τη σύνθεση ρόλων, $\mathcal{R}$.

Οι σχέσεις μεταξύ των εννοιών με τη μορφή αξιωμάτων συνθέτουν το *σώμα ορολογίας* (*TBox*) ή αλλιώς TBox (Terminological Box), $\mathcal{T}$. Οι ΠΛ επιτρέπουν επίσης να καθορίσουμε τις σχέσεις των ατόμων με έννοιες, ή ρόλους με τους *ισχυρισμούς* (*assertions*). Έτσι, δηλώνουμε ότι η Άννα είναι φοιτήτρια με τον ισχυρισμό Φοιτητής(Άννα). Το σύνολο των ισχυρισμών ορίζει το *σώμα ισχυρισμών* (*assertional component*), ή αλλιώς ABox (Assertion Box) μιας ΠΛ και συμβολίζεται με $\mathcal{A}$. Το σώμα ορολογίας, TBox μαζί με το σώμα ισχυρισμών, ABox αποτελούν μια *Βάση Γνώσης (ΒΓ)* (*Knowledge Base – KB*) που περιγράφει ένα συγκεκριμένο επιστημονικό πεδίο ή εφαρμογή. Το σώμα ορολογίας TBox αναφέρεται και ως *οντολογία* (*ontology*).

Η σημασιολογία μιας γλώσσας ΠΛ δίνεται από την *ερμηνεία* (*interpretation*) της $\mathcal{I}$, επί ενός πεδίου $\Delta^{\mathcal{I}}$. Μία ερμηνεία αντιστοιχίζει τα άτομα της ΠΛ σε αντικείμενα του πεδίου $\Delta^{\mathcal{I}}$, τις έννοιες σε υποσύνολά του, και τους ρόλους σε σύνολα από ζεύγη αντικειμένων του πεδίου. Η καθολική έννοια $\top$ αντιστοιχίζεται στο $\Delta^{\mathcal{I}}$ και η κενή έννοια $\bot$ στο κενό σύνολο. Τα αξιώματα και οι ισχυρισμοί ερμηνεύονται με τον προφανή τρόπο και έτσι για παράδειγμα το αξίωμα Μεταπτυχιακός $\sqsubseteq$ Φοιτητής δηλώνει ότι οι μεταπτυχιακοί φοιτητές είναι υποσύνολο του συνόλου των φοιτητών, $(\text{Μεταπτυχιακός} \sqsubseteq \text{Φοιτητής})^{\mathcal{I}} = \text{Μεταπτυχιακός}^{\mathcal{I}} \subseteq \text{Φοιτητής}^{\mathcal{I}}$. Ομοίως, σύμφωνα με ερμηνεία ενός ισχυρισμού Φοιτητής(Άννα), η ερμηνεία του ατόμου 'Αννα, δηλαδή το αντικείμενο $\text{Άννα}^{\mathcal{I}}$ ανήκει στο σύνολο $\text{Φοιτητής}^{\mathcal{I}}$. Μια ερμηνεία που ικανοποιεί όλα τα αξιώματα του TBox καθώς και όλους τους ισχυρισμούς του ABox λέγεται *μοντέλο* (*model*) της βάσης γνώσης.

1.2 Προβλήματα Συλλογιστικής

Οι πληροφορίες και η γνώση που σχετίζονται με ένα πεδίο εφαρμογής μπορούν να αναπαρασταθούν τυπικά, με γλώσσες ΠΛ και να δομηθούν σε οντολογίες. Απώτερος σκοπός είναι η αυτοματοποιημένη επεξεργασία της γνώσης με διεργασίες συλλογιστικής για την εξαγωγή συμπερασμάτων. Ένα από τα βασικά προβλήματα συλλογιστικής είναι ο έλεγχος *ικανοποιησιμότητας* (*satisfiability*) της βάσης γνώσης, όπου εξετάζεται εάν η βάση γνώσης διαθέτει μοντέλο. Υπάρχουν πολλά προβλήματα συλλογιστικής που μπορούν να αναχθούν στον έλεγχο ικανοποιησιμότητας όπως για παράδειγμα, η *υπαγωγή* (*subsumption*) εννοιών. Πιο συγκεκριμένα, ο έλεγχος εαν μία έννοια, C_1, υπάγεται σε μία άλλη, C_2, ανάγεται στον έλεγχο (μη) ικανοποιησιμότητας της βάσης γνώσης με σώμα ορολογίας $\{A \sqsubseteq C_1, A \sqsubseteq \neg C_2\}$ και σώμα ισχυρισμών $\{A(a)\}$. Παρόμοια, η *ικανοποιησιμότητα* μίας *έννοιας* C ανάγεται στον έλεγχο (μη) ικανοποιησιμότητας μιας βάσης γνώσης με σώμα ισχυρισμών $\{C(a)\}$.

Στην παρούσα εργασία εστιάζουμε το ενδιαφέρον μας στον πρόβλημα της *απάντησης ερωτημάτων* (*query answering*) με βάση κάποια οντολογία. Το πρόβλημα αυτό αφορά εφαρμογές όπου η γνώση του σχετικού πεδίου περιγράφεται από κάποια οντολογία, ενώ οι χρήστες της εφαρμογής διατυπώνουν ερωτήματα χρησιμοποιώντας όρους της οντολογίας. Οι απαντήσεις στα ερωτημάτα ανακλούν τόσο τα αποθηκευμένα δεδομένα, όσο και τα αξιώματα της οντολογίας.

1.2.1 Απάντηση Ερωτημάτων

Το πρόβλημα που θα μας απασχολήσει στην εργασία μας αφορά την απάντηση *Συζευκτικών Ερωτημάτων, ΣΕ* (*Conjuctive Queries, CQ*) με βάση οντολογίες εκφρασμένες σε γλώσσες ΠΛ. Τα συζευκτικά ερωτήματα είναι εκφραστικά, σύνθετα ερωτήματα, ιδιαίτερα δημοφιλή στον κλάδο των βάσεων δεδομένων [1]. Σαν παράδειγμα του προβλήματος που μας απασχολεί, θεωρούμε τη βάση γνώσης με σώμα ορολογίας $\mathcal{T}$ και σώμα ισχυρισμών $\mathcal{A}$:

$$\mathcal{T} = \{\textsf{Μεταπτυχιακός} \sqsubseteq \textsf{Φοιτητής} \sqcap \exists\textsf{έχειΠτυχίο}\}$$

$$\mathcal{A} = \{\textsf{Μεταπτυχιακός}(\textsf{Άννα}), \textsf{έχειΠτυχίο}(\textsf{Τάσος}), \textsf{Φοιτητής}(\textsf{Γιάννης}), \textsf{έχειΠτυχίο}(\textsf{Γιάννης})\}$$

Έστω ότι επιθυμούμε να αναζητήσουμε τους φοιτητές της βάσης γνώσης που έχουν πτυχίο. Διατυπώνουμε το παρακάτω συζευκτικό ερώτημα:

$$Q(x) \leftarrow \textsf{Φοιτητής}(x) \wedge \textsf{έχειΠτυχίο}(x)$$

Λαμβάνοντας υπόψη το σώμα ορολογίας, το ερώτημα θα πρέπει να μας επιστρέψει σαν απάντηση τους φοιτητές που έχουν πτυχίο, ή τους μεταπτυχιακούς φοιτητές. Τυπικά, η απάντηση του ερωτήματος αποτελείται από όλα τα άτομα της βάσης γνώσης με τα οποία μπορούμε να αντικαταστήσουμε τη μεταβλητή x, ώστε η πρόταση που προκύπτει να είναι αληθής σε οποιοδήποτε μοντέλο της βάσης γνώσης. Στο παράδειγμά μας, για το δεδομένο σώμα ισχυρισμών, οι απάντησεις είναι τα άτομα Άννα, Γιάννης.

Οι αυξανόμενες απαιτήσεις των σύγχρονων εφαρμογών για αποτελεσματική διαχείρηση μεγάλου όγκου πληροφορίας και σημασιολογική αναπαράσταση των δεδομένων, οδήγησαν στη μελέτη του προβλήματος της αναπαράστασης της γνώσης ενός πεδίου με οντολογίες, καθώς και της απάντησης ερωτημάτων με βάση τις οντολογίες [3, 98, 111, 36, 50]. Ο κλάδος της *Πρόσβασης σε Δεδομένα πάνω από Οντολογίες* (*Ontology Based Data Access, OBDA*) [3, 42, 60] διαπραγματεύεται το πρόβλημα της απάντησης ερωτημάτων σε εφαρμογές όπου η γνώση του πεδίου περιγράφεται από κάποια οντολογία. Τα δεδομένα της εφαρμογής είναι αποθηκευμένα σε μία βάση δεδομένων, ή σε ένα σώμα ισχυρισμών, ενώ οι σχέσεις και οι ιδιότητές τους περιγράφονται από τα αξιώματα της οντολογίας. Έτσι, σε σενάρια όπου τα δεδομένα της εφαρμογής συλλέγονται από διαφορετικές πηγές, τα αξιώματα της οντολογίας διασφαλίζουν την ακεραιότητα της βάσης δεδομένων. Επιπλέον, ο χρήστης της εφαρμογής έχει τη δυνατότητα να διατυπώνει ερωτήματα χρησιμοποιώντας το λεξιλόγιο της οντολογίας. Κατά την απάντηση του ερωτήματος του χρήστη, πραγματοποιούνται διαδικασίες συλλογιστικής που αξιοποιούν τη διαθέσιμη γνώση, και την ενσωματώνουν στο ερώτημα. Με αυτόν τον τρόπο, στην περίπτωση που η βάση δεδομένων δεν είναι πλήρης, διασφαλίζεται ότι θα ανακτηθούν όλες οι απαντήσεις που ικανοποιούν το ερώτημα και τη δεδομένη οντολογία.

Το πρόβλημα της απάντησης των ερωτημάτων σε οντολογίες έχει μελετηθεί σε πληθώρα ερευνητικών εργασιών, που προτείνουν διαφορετικές προσεγγίσεις και μεθόδους, ενώ ακολούθησαν ποικίλες πρακτικές υλοποιήσεις [67, 44, 130, 123, 135, 54, 79, 84]. Παρά την εκτενή έρευνα, τα συστήματα απάντησης δεν ανταποκρίνονται ικανοποιητικά σε περιπτώσεις μεγάλου όγκου δεδομένων λόγω της υψηλής υπολογιστικής πολυπλοκότητας του προβλήματος [103, 53, 89, 69]. Πράγματι, η πολυπλοκότητα *ως προς τα δεδομένα* (*data complexity*) σε σχέση δηλαδή με το μέγεθος του ABox, σε περιπτώσεις πολύ εκφραστικών περιγραφικών λογικών, είναι co-NP-complete [90]. Προκειμένου οι αλγόριθμοι απάντησης συζευκτικών ερωτημάτων να είναι πρακτικοί, και να αποκρίνονται σε πολυωνυμικό χρόνο, αναπτύχθηκαν οι *βατές* (*tractable*) γλώσσες περιγραφικής λογικής, όπως είναι η οικογένεια γλωσσών DL-Lite [37, 32, 33], η $\mathcal{EL}$ [5, 83], η οικογένεια της Datalog$^{\pm}$ [27, 26, 28] και η Horn-$\mathcal{SHIQ}$ [69, 43, 45, 76, 86].

Απάντηση Ερωτημάτων με Επαναγραφή

Η μέθοδος της *επαναγραφής* (*rewriting*) αποτελεί μία σημαντική προσέγγιση για την απάντηση ερωτημάτων σε οντολογίες [33]. Σύμφωνα με την προσέγγιση αυτή το δεδομένο ερώτημα και η οντολογία μετασχηματίζονται, ώστε να αναπαρασταθούν με φορμαλισμούς που είναι διαχειρίσιμοι από συστήματα ανάκτησης δεδομένων [56, 71, 33, 109, 110, 95]. Σαν παράδειγμα απάντησης ερωτήματος με επαναγραφή, ας θεωρήσουμε μία οντολογία που χρησιμοποιείται για να περιγράψει τα δεδομένα ενός πανεπιστημίου και περιλαμβάνει τα παρακάτω αξιώματα:

$$\mathcal{T} = \{\text{Μεταπτυχιακός} \sqsubseteq \text{Φοιτητής}, \text{Προπτυχιακός} \sqsubseteq \text{Φοιτητής}\}$$

Έστω ότι διατυπώνουμε ένα ερώτημα με σκοπό προκειμένου να αναζητήσουμε τους καταχωρημένους φοιτητές στη βάση του πανεπιστημίου:

$$q_1 : \quad Q(x) \leftarrow \textsf{Φοιτητής}(x)$$

Σύμφωνα με την τεχνική της επαναγραφής, διατυπώνονται νέα ερωτήματα με βάση το αρχικό λαμβάνοντας υπόψη τα αξιώματα του σώματος ορολογίας. Έτσι, προκύπτουν τα παρακάτω:

$$q_2 : \quad Q(x) \leftarrow \textsf{Προπτυχιακός}(x), \quad q_3 : \quad Q(x) \leftarrow \textsf{Μεταπτυχιακός}(x)$$

Επομένως, το σώμα ορολογίας και το αρχικό ερώτημα έχουν γραφτεί εκ νέου, στο σύνολο που περιλαμβάνει τα $\{q_1, q_2, q_3\}$ και ονομάζουμε επαναγραφή. Κατόπιν, η ανάκτηση των δεδομένων από το σώμα ισχυρισμών γίνεται αξιοποιώντας το σύνολο της επαναγραφής.

Η δυνατότητα απάντησης συζευκτικού ερωτήματος με τη μέθοδο της επαναγραφής προκύπτει από την παρατήρηση ότι η οντολογία μαζί με το ερώτημα ανάγονται σε ένα πρόγραμμα *διαζευκτικής Datalog* (*disjunctive Datalog*), το οποίο συνεπάγεται το ίδιο σύνολο από *βασικά γεγονότα* (*ground facts*) με την οντολογία. Επομένως, το datalog πρόγραμμα, ή αλλιώς επαναγραφή, μπορεί να αποτιμηθεί σε ένα οποιοδήποτε σύνολο δεδομένων, αγνοώντας την αρχική οντολογία. Βασικό πλεονέκτημα της μεθόδου είναι ότι επιτρέπει την αξιοποίηση τεχνολογιών *επαγωγικών βάσεων δεδομένων* (*deductive databases*) [38] για την ανάκτηση των απαντήσεων. Μία από τις πρώτες μελέτες προς αυτή την κατεύθυνση έγινε στο [96], όπου περιγράφεται η δυνατότητα μετασχηματισμού μιας οντολογίας διατυπωμένης στη γλώσσα $\mathcal{SHIQ}$, σε ένα ισοδύναμο πρόγραμμα datalog. Η συγκεκριμένη εργασία οδήγησε στην ανάπτυξη του συστήματος KAON2 [99], που όμως, σημειώνουμε ότι δεν υποστηρίζει την επαναγραφή συζευκτικών ερωτημάτων. Σε περισσότερο πρόσφατες εργασίες [127, 126], το datalog πρόγραμμα της επαναγραφής δεν αποτιμάται άμεσα, αλλά η επαναγραφή της οντολογίας αξιοποιείται από μεθόδους απάντησης ερωτημάτων που βασίζονται σε τεχνολογίες του σημασιολογικού ιστού. Σημειώνουμε επίσης, τις προσεγγίσεις που εφαρμόζουν τη *συνδυαστική μέθοδο* (*combined approach*) [81, 91]. Σύμφωνα με τη μέθοδο αυτή, το δεδομένο σώμα ισχυρισμών της βάσης γνώσης επεκτείνεται βάσει της οντολογίας και στη συνέχεια, μία μερική επαναγραφή του ερωτήματος αποτιμάται στο επεκταμένο σύνολο δεδομένων.

Η τεχνική της επαναγραφής έχει μελετηθεί για διαφορετικές εκφραστικότητες γλωσσών περιγραφικής λογικής. Πλέον, η μέθοδος αποτελεί τη συνηθέστερη, ενδεχομένως, πρακτική απάντησης ερωτημάτων με βάση κάποια οντολογία, για γλώσσες ΠΛ χαμηλότερης εκφραστικότητας όπως είναι η DL-Lite [33, 57]. Βασική ιδιότητα της DL-Lite είναι η *δυνατότητα επαναγραφής σε ΛΠΤ* (*FO-rewritability*), που υπονοεί ότι η επαναγραφή ενός συζευκτικού ερωτήματος με βάση την οντολογία είναι ένα σύνολο συζευκτικών ερωτημάτων. Σημειώνουμε επίσης, ότι υποσύνολα της Datalog$^{\pm}$ [26, 29] χαρακτηρίζονται από τη δυνατότητα επαναγραφής σε ΛΠΤ. Στις περιπτώσεις αυτές, η αποτίμηση της επαναγραφής ανήκει στην κλάση πολυπλοκότητας LOGSPACE, και πιο συγκεκριμένα στην AC_0 [106, 131, 33]. Έχουν αναπτυχθεί πολλά πρακτικά

συστήματα που υλοποιούν τη μέθοδο επαναγραφής και υποστηρίζουν την DL-Lite, Mastro [34], Presto [116], Quest [113], Rapid [40], IQAROS [132], και Ontop [92], και το Nyaya [102] που υποστηρίζει τη γραμμική Datalog$^{\pm}$.

Οι γλώσσες υψηλότερης εκφραστικότητας $\mathcal{ELHI}$ και Horn-$\mathcal{SHIQ}$, δε διαθέτουν τη δυνατότητα επαναγραφής σε ΛΠΤ και έτσι, η επαναγραφή έχει πάντοτε τη μορφή datalog προγράμματος. Η πολυπλοκότητα (με βάση τα δεδομένα) για την αποτίμηση της επαναγραφής είναι PTIME-complete [106, 131, 114, 82, 84]. Σημειώνουμε ότι για τη γλώσσα $\mathcal{ELHI}$ έχει αναπτυχθεί το σύστημα το Requiem [110], το kyrie2 [94], και για την Horn-$\mathcal{SHIQ}$ το Clipper [45].

Επαναγραφή με τη Μέθοδο Επίλυσης

Μία σημαντική προσέγγιση για την τεχνική επαναγραφής είναι η μέθοδος της *επίλυσης* (*resolution*) [13]. Σύμφωνα με την προσέγγιση αυτή, η οντολογία και το ερώτημα που διατυπώνεται επί αυτής, μετατρέπονται σε ένα σύνολο προτάσεων λογικής πρώτης τάξης στις οποίες εφαρμόζεται εξαντλητικά ο κανόνας της επίλυσης. Για παράδειγμα, θεωρούμε τον παρακάτω κανόνα επίλυσης, που εφαρμόζεται για την κατασκευή της επαναγραφής στο παράδειγμα της προηγούμενης ενότητας:

$$\frac{Q(x) \leftarrow \textsf{Φοιτητής}(x), \quad \textsf{Φοιτητής}(x) \leftarrow \textsf{Προπτυχιακός}(x)}{Q(x) \leftarrow \textsf{Προπτυχιακός}(x)}$$

Έχει αναπτυχθεί πληθώρα μεθόδων που εφαρμόζουν την μέθοδο της επίλυσης για υποσύνολα της λογική πρώτης τάξης, και αφορούν προβλήματα συλλογιστικής περιγραφικών λογικών [85, 75, 58]. Όπως έχουμε αναφέρει, το πρώτο σύστημα απάντησης ερωτημάτων που υλοποιεί την μεθόδο επαναγραφής και βασίζεται στην επίλυση, είναι το KAON2 [68, 99], το οποίο υποστηρίζει τη γλώσσα $\mathcal{SHIQ}$ και Horn-$\mathcal{SHIQ}$. Το σύστημα KAON2 εφαρμόζει κανόνες *βασικής υπέρθεσης* (*basic superposition*) [12], *ταξινομημένης υπερεπίλυσης* (*ordered hyperresolution*), και διαχειρίζεται *βασικά συζευκτικά ερωτήματα* (*ground conjunctive queries*). Αργότερα, προτάθηκε το σύστημα Requiem [110] που διαχειρίζεται οντολογίες διατυπωμένες στις γλώσσες DL-Lite και $\mathcal{ELHI}$. Ο αλγόριθμος του συστήματος εφαρμόζει μια από-πάνω-προς-τα-κάτω SLD επίλυση [88]. Στην εργασία [102] περιγράφεται ένας αλγόριθμος που εφαρμόζει την επίλυση για τον υπολογισμό της επαναγραφής στις *γραμμική Datalog*$^{\pm}$ (*linear Datalog*$^{\pm}$) και DL-Lite. Ο αλγόριθμος υλοποιείται στο σύστημα απάντησης ερωτημάτων Nyaya [41].

Η τεχνική της επίλυσης κατά την επαναγραφή των ερωτημάτων, είναι ιδιαίτερα χρήσιμη καθώς επιτρέπει την υιοθέτηση πληθώρας βελτιστοποιήσεων που έχουν προταθεί στη θεωρία αποδείξεων της λογικής πρώτης τάξης [13]. Ένας αλγορίθμος επαναγραφής ερωτημάτων που βασίζεται την τεχνική της επίλυσης μπορεί με ευκολία να επεκταθεί ώστε να υποστηρίζει γλώσσες ΠΛ υψηλότερης εκφραστικότητας και να αξιοποιήσει υπάρχουσες τεχνικές βελτιστοποίησης. Παρά τα πλεονεκτήματά της όμως, η τεχνική διαθέτει χαρακτηριστικά που δυσχεραίνουν την αποτελεσματικότητα των συστημάτων που την εφαρμόζουν [122]. Το βασικό μειονέκτημα της μεθόδου είναι ότι είναι γενικού σκοπού. Έτσι, ένας αλγόριθμος επαναγραφής που βασίζεται

στην επίλυση, εφαρμόζει τους κανόνες της εξαντλητικά, με μία από-πάνω-προς-τα-κάτω προσέγγιση, με αποτέλεσμα σε πολλές περιπτώσεις την παραγωγή προτάσεων που δε συμμετέχουν στην παραγωγή της επαναγραφής και απορρίπτονται στην έξοδο του συστήματος.

1.3 Ανοικτά προβλήματα και προσέγγιση

Το πρόβλημα της απάντησης των ερωτημάτων με βάση οντολογίες αποτελεί ένα ιδαίτερα απαιτητικό πρόβλημα για τις σύγχρονες εφαρμογές που καλούνται να διαχειριστούν με αποτελεσματικό τρόπο μεγάλο όγκο δεδομένων [79, 98, 111]. Η υψηλή υπολογιστική πολυπλοκότητα του προβλήματος [103, 53, 89, 78, 35] και το πλήθος των δεδομένων παίζει καθοριστικό ρόλο στην επίδοση των συστημάτων απάντησης ερωτημάτων. Συστήματα που εφαρμόζουν άμεσα τεχνικές tableaux, όπως είναι τα HermiT [130], Pellet [123], and Racer [135], παρά τις σημαντικές βελτιστοποιήσεις που υλοποιούν, παρουσιάζουν αδυναμίες στην πράξη καθώς αυξάνεται το πλήθος των δεδομένων, και συχνά δεν καταφέρνουν να αποκριθούν.

Οι αδυναμίες των αλγορίθμων tableaux οδήγησαν στην μελέτη διαφορετικών μηχανισμών απάντησης ερωτημάτων, που στοχεύουν στον υπολογισμό της επαναγραφής του ερωτήματος με βάση την οντολογία. Αρκετές από αυτές τις προσεγγίσεις υιοθετούν τη μέθοδο της επίλυσης [13]. Σε αυτό το πλαίσιο, η οντολογία της εισόδου και το ερώτημα μετατρέπονται σε ένα σύνολο προτάσεων όπου εφαρμόζεται εξαντλητικά ο κανόνας της επίλυσης. Αποτέλεσμα της διαδικασίας είναι η παραγωγή ενός συνόλου προτάσεων, ορισμένες από τις οποίες περιλαμβάνουν συναρτησιακά σύμβολα και πρέπει να εξαιρεθούν για την κατασκευή της επαναγραφής με τη μορφή datalog προγράμματος.

Η πρώτη προσπάθεια αξιοποίησης της μεθόδου της επίλυσης σε μηχανισμούς συλλογιστικής για περιγραφικές λογικές έγινε στις εργασίες [96, 85, 75]. Πρόκειται για μία σημαντική προσπάθεια εφαρμογής της μεθόδου σε εκφραστικά υποσύνολα της λογικής πρώτης τάξης. Από την προσέγγιση αυτή, προκύπτει ο τρόπος με τον οποίο μία οντολογία σε ΠΛ μετατρέπεται σε ένα ισοδύναμο πρόγραμμα datalog. Το προκύπτον σύνολο από datalog προτάσεις μπορεί στη συνέχεια, να αξιοποιηθεί για την απάντηση ενός συζευκτικού ερωτήματος με βάση την οντολογία. Ένα άμεσο όφελος που προκύπτει από αυτή την προσέγγιση, είναι η δυνατότητα αξιοποίησης της πληθώρας των τεχνικών που έχουν αναπτυχθεί στον κλάδο των βάσεων δεδομένων. Επίσης, σημαντικό πλεονέκτημα της μεθόδου είναι ότι χαρακτηρίζεται από δυνατότητα επεκτασης, ώστε να καλύπτει περιγραφικές λογικές υψηλής εκφραστικότητας. Έτσι, ένας αλγορίθμος επαναγραφής που βασίζεται στην επίλυση, μπορεί με ευκολία να επεκταθεί ώστε να υποστηρίζει γλώσσες ΠΛ υψηλότερης εκφραστικότητας. Επιπλέον, μπορεί να αξιοποιήσει υπάρχουσες τεχνικές βελτιστοποίησης της μεθόδου, που έχουν προταθεί για διαφορετικά υποσύνολα λογικής πρώτης τάξης [47, 72].

Η μέθοδος της επίλυσης αποτελεί μία σημαντική προσέγγιση για την επαναγραφή οντολογιών, έχουν όμως αναπτυχθεί αρκετά πρακτικά συστήματα επαναγραφής που δεν εφαρμόζουν τους κανόνες της επίλυσης, και είναι ειδικά σχεδιασμένα για κάποια

από τις βατές γλώσσες περιγραφικής λογικής. Συγκεκριμένα, στο [33] έχει προταθεί ο αλγόριθμος επαναγραφής PerfectRef, για την οικογένεια γλωσσών της DL-Lite. Ο αλγόριθμος κατασκευάζει την επαναγραφή ενός συζευκτικού ερωτήματος με βάση μία DL-Lite οντολογία, στη μορφή ενός συνόλου συζευκτικών ερωτημάτων. Από την μελέτη του αλγορίθμου προκύπτει ότι η πολυπλοκότητα για την επαναγραφή σε DL-Lite είναι AC. Πρακτικά συστήματα, όπως το QuOnto[2] και Owlgres[125], υλοποιούν το συγκεκριμένο αλγόριθμο.

Στη συνέχεια, στο [115] παρουσιάζεται ένας αλγόριθμος επαναγραφής για την $\mathcal{ELH}$ [6], που επιστρέφει ένα βέλτιστο datalog πρόγραμμα. Από την μελέτη του αλγορίθμου προκύπτει ότι στην περίπτωση της $\mathcal{ELH}$, η πολυπλοκότητα απάντησης συζευκτικού ερωτήματος με βάση τα δεδομένα είναι PTime-complete.

Ένας ιδιαίτερα βελτιστοποιημένος αλγόριθμος επαναγραφής για την DL-Lite περιγράφεται στην εργασία [116]. Πρόκειται για τον αλγόριθμο Presto ο οποίος έχει σκοπό να αποφύγει το εκθετικό μέγεθος της ΣΣΕ επαναγραφής, ως προς το πλήθος των ατόμων του ερωτήματος, με την κατασκευή ενός μη-αναδρομικού datalog προγράμματος.

Μία οικογένεια αλγορίθμων έχει αναπτυχθεί για τις γλώσσες Datalog$^{\pm}$. Στις εργασίες [31, 55] περιγράφεται μία οικογένεια αλγορίθμων επαναγραφής για τις γλώσσες Datalog$^{\pm}$. Ο αλγόριθμος [31] βασίζεται στον [33] για την DL-Lite, ενώ μία βελτιστοποίηση του αλγορίθμου εισάγεται στο [55], όπου εφαρμόζονται τεχνικές που αποτρέπουν την κατασκευή περιττών ερωτημάτων που υπάγονται σε άλλα. Οι αλγόριθμοι υλοποιούνται στο σύστημα [41].

Στην εργασία [45] παρουσιάζεται ένας αλγόριθμος επαναγραφής για την Horn-$\mathcal{SHIQ}$ [86, 105]. Ο αλγόριθμος υπολογίζει την επαναγραφή συζευκτικού ερωτήματος εφαρμόζοντας τα αξιώματα του σώματος ορολογίας στο ερώτημα, και επιστρέφει ένα datalog πρόγραμμα. Υλοποιείται στο σύστημα Clipper, το μοναδικό εως τώρα, σύστημα επαναγραφής συζευκτικού ερωτήματος για τη Horn-$\mathcal{SHIQ}$.

Τέλος, στην εργασία [132] παρουσιάστηκε ένας βελτιστοποιημένος αλγόριθμος επαναγραφής συζευκτικού ερωτήματος για DL-Lite οντολογίες. Πρόκειται για έναν αλγόριθμο επαυξητικής επαναγραφής [133], που επεξεργάζεται το κάθε άτομο ξεχωριστά, και εισάγει πληθώρα τεχνικών βελτιστοποίησης κατά τον υπολογισμό της τελικής επαναγραφής του ερωτήματος. Ο αλγόριθμος υλοποιήθηκε στο σύστημα IQAROS.

Τα προαναφερθέντα συστήματα εξειδικεύονται σε συγκεκριμένες γλώσσες, ενώ δεν είναι προφανές πώς θα μπορούσαν να επεκταθούν, ώστε να υποστηρίζουν πιο εκφραστικές περιγραφικές λογικές. Την ίδια στιγμή, οι αλγόριθμοι που βασίζονται στην επίλυση [99, 110] μπορούν να προσαρμοστούν σε διαφορετικά υποσύνολα λογικής πρώτης τάξης. Παρά τη δυνατότητα επέκτασης όμως, οι αλγόριθμοι επαναγραφής που εφαρμόζουν τους κανόνες τις επίλυσης, παρουσιάζουν ορισμένες αδυναμίες που ενδέχεται να τους καταστήσουν μη αποδοτικούς στην πράξη [122]. Συγκεκριμένα, οι αλγόριθμοι που βασίζονται στην επίλυση είναι γενικού σκοπού, εφαρμόζουν τους κανόνες της επίλυσης εξαντλητικά, χωρίς καθορισμένη στρατηγική. Η μη ελεγχόμενη εφαρμογή των κανόνων έχει ως αποτέλεσμα την παραγωγή μεγάλου πλήθους προτάσεων που δεν συμπεριλαμβάνονται απαραίτητα στην έξοδο του αλγορίθμου. Πράγματι,

διαφορετικά μονοπάτια παραγωγής, ενδέχεται να παράγουν την ίδια πρόταση, ενώ συχνά, παράγονται σε προτάσεις με συναρτησιακά σύμβολα τα οποία δεν είναι δυνατό να απομακρυνθούν. Οι προτάσεις με συναρτήσεις δεν συμπεριλαμβάνονται στο τελικό πρόγραμμα datalog, ενώ την ίδια στιγμή, ενδέχεται να μη συμβάλλουν στην παραγωγή άλλων που συμπεριλαμβάνονται στο σύνολο της επαναγραφής. Όπως γίνεται κατανοητό, η επίδοση των αλγορίθμων επαναγραφής μπορεί να επιβαρυνθεί σημαντικά απο τα μακροσκελή μονοπάτια παραγωγής, και την κατασκευή προτάσεων που δεν είναι χρήσιμες ως προς την επαναγραφή. Στην πράξη, και ιδιαίτερα στις περπτώσεις οντολογιών μεγάλης κλίμακας [73], οι σχετικοί αλγόριθμοι γενικού σκοπού δεν είναι αποδοτικοί.

Σκοπός της εργασίας μας είναι ο σχεδιασμός ενός βελτιστοποιημένου αλγόριθμου επαναγραφής που βασίζεται στην τεχνική της επίλυσης. Ο αλγόριθμός μας υποστηρίζει εκφραστικά υποσύνολα της γλώσσας OWL αξιοποιώντας τη δυνατότητα γενίκευσης της τεχνικής, ενώ την ίδια στιγμή, αντιμετωπίζει τη βασική αδυναμία της μεθόδου κατά τον υπολογισμό της επαναγραφής, εφαρμόζοντας τους κανόνες της επίλυσης μόνο όταν πρόκειται να παράγουν προτάσεις χωρίς συναρτησιακά σύμβολα.

Για το σκοπό αυτό αναπτύξαμε έναν πρωτότυπο αλγόριθμο για την DL-Lite, που εφαρμόζει τους κανόνες της επίλυσης με ελεγχόμενο τρόπο, απλουστεύοντας τη διαδικασία επαναγραφής έτσι όπως παρουσιάζεται στο [108]. Πιο συγκεκριμένα, ο αλγόριθμός μας παράγει ενδιάμεσες προτάσεις που περιέχουν συναρτησιακά σύμβολα μόνο όταν είναι γνωστό εκ των προτέρων, ότι πρόκειται να χρησιμοποιηθούν κατά την παραγωγή νέων προτάσεων που θα συμπεριληφθούν στο αποτέλεσμα της επαναγραφής. Κάτι τέτοιο επιτυγχάνεται με ένα νέο κανόνα επίλυσης που ονομάζουμε *κανόνα συρρίκνωσης* (*shrinking*). Ο κανόνας της συρρίκνωσης συγχωνεύει περισσότερα από ένα βήματα επίλυσης εισάγοντας ταυτόχρονα συγκεκριμένους περιορισμούς στην παραγόμενη πρόταση. Πέρα από την ελεγχόμενη εφαρμογή των κανόνων της επίλυσης, ο αλγόριθμός μας εισάγει επιπλέον τεχνικές που αφορούν την κατασκευή βέλτιστου τελικού datalog προγράμματος, με κριτήριο το πλήθος των προτάσεων που περιλαμβάνει. Μάλιστα, το σύνολο των βελτιστοποιήσεων που εφαρμόζεται αποτρέπει την κατασκευή προτάσεων που υπάγονται σε άλλες, με αποτέλεσμα μία ακόμα περισσότερο περιορισμένη εφαρμογή κανόνων.

Στη συνέχεια, ο αλγόριθμος επεκτείνεται ώστε να καλύψει την εκφραστικότητα της γλώσσας $\mathcal{ELHI}$. Πρόκειται για μία ενδιαφέρουσα περίπτωση γλώσσας περιγραφικής λογικής λόγω των τεχνικών δυσκολιών που ανακύπτουν από τη μορφή των αξιώματων που υποστηρίζει, και τον τρόπο που αυτά μπορούν να συνδυαστούν κατά την εφαρμογή των κανόνων της επίλυσης. Προκειμένου να αντιμετωπίσουμε τις σύνθετες αλληλεπιδράσεις μεταξύ των αξιωμάτων στην $\mathcal{ELHI}$, επεκτείνουμε τον κανόνα της συρρίκνωσης όπως ορίζεται για την περιπτωση της DL-Lite, ώστε να εφαρμόζεται χρησιμοποιώντας περισσότερες προτάσεις. Επιπλέον, προτείνουμε ένα νέο κανόνα που ονομάζουμε *κανόνα συνάρτησης* (*function-rule*).

Τέλος, ο αλγόριθμος που προτείνουμε υποστηρίζει την υψηλής εκφραστικότητας γλώσσα, Horn-$\mathcal{SHIQ}$. Ο κανόνας της συρρίκνωσης προσαρμόζεται κατάλληλα, ώστε να λαμβάνει υπόψη τις προτάσεις με ισότητα που περιλαμβάνει η συγκεκριμένη γλώσσα. Επιπλέον, ακολουθώντας τη βασική ιδέα των αλγορίθμων για τις DL-Lite

και $\mathcal{ELHI}$, ο αλγόριθμος αποφεύγει την επίλυση προτάσεων με ισότητα με άλλες, όταν είναι γνωστό εκ των προτέρων, ότι πρόκειται να παραχθούν προτάσεις που διαθέτουν συναρτησιακά σύμβολα και δε συνεισφέρουν στο αποτέλεσμα της τελικής επαναγραφής. Για το σκοπό αυτό, προτείνουμε δύο νέους κανόνες που συγχωνεύουν περισσότερα βήματα επίλυσης και εισάγουν περιορισμούς στην παραγόμενη πρόταση. Ονομάζουμε τους κανόνες αυτούς *κανόνα υπέρθεσης-1* (*superosition-1*), *κανόνα υπέρθεσης-2* (*superposition-2*). Σημειώνουμε ότι ο αλγόριθμός μας αποτελεί την πρώτη προσπάθεια βελτιστοποίησης της μεθόδου της επίλυσης σε υποσύνολο της λογικής πρώτης τάξης που υποστηρίζει προτάσεις με ισότητα.

Ο αλγόριθμος υλοποιήθηκε στο σύστημα επαναγραφής Rapid [40, 128]. Η πειραματική αξιολόγηση του συστήματος Rapid έγινε χρησιμοποιώντας μεγάλης κλίμακας οντολογίες του πραγματικού κόσμου που αφορούν την DL-Lite, την $\mathcal{ELHI}$ καθώς και την Horn-$\mathcal{SHIQ}$. Η σύγκριση του Rapid με τα συστήματα που βρίσκονται στην αιχμή της τεχνολογίας έδειξε ότι σε περιπτώσεις σύνθετων οντολογιών με μεγάλο όγκο αξιωμάτων τα υπάρχοντα συστήματα δεν μπορούν να ανταπεξέλθουν, και συχνά, δεν τερματίζουν νωρίτερα από τις τρεις ώρες, ενώ το Rapid τερματίζει σε μερικά δευτερόλεπτα. Μάλιστα, στην περίπτωση της Horn-$\mathcal{SHIQ}$, το μοναδικό υπάρχον σύστημα επαναγραφής δεν καταφέρνει να τερματίσει όταν το σύστημά μας αποκρίνεται σε λίγα δευτερόλεπτα. Τα αποτελέσματα που προκύπτουν από την αξιολόγηση του συστήματος είναι ιδιαίτερα ενθαρρυντικά.

1.4 Δομή της εργασίας

Η παρούσα εργασία οργανώνεται ως εξής:

- Στη δεύτερη ενότητα παρουσιάζουμε το θεωρητικό υπόβαθρο που είναι απαραίτητο για την κατανόηση της εργασίας. Συγκεκριμένα, παρουσιάζουμε τις γλώσσες ΠΛ DL-Lite, $\mathcal{ELHI}$ και Horn-$\mathcal{SHIQ}$. Στη συνέχεια, υπενθυμίζουμε έννοιες από τη λογική πρώτης τάξης, τη μέθοδο της επίλυσης, και δίνουμε τους ορισμούς του συζευκτικού ερωτήματος και της επαναγραφής. Τέλος, περιγράφουμε τους αλγόριθμους που υλοποιούνται στα συστήματα Requiem και KAON2, καθώς αποτέλεσαν σημείο αφετηρίας για το σχεδιασμό του αλγόριθμου μας στην περίπτωση της $\mathcal{ELHI}$ και Horn-$\mathcal{SHIQ}$ αντίστοιχα.

- Στην τρίτη ενότητα περιγράφουμε τον αλγόριθμο επαναγραφής που προτείνουμε για την DL-Lite. Σημειώνουμε τα μειονεκτήματα των αλγορίθμων που βασίζονται στην επίλυση και εξηγούμε πώς ο αλγόριθμός μας τα αντιμετωπίζει. Τέλος, αποδεικνύουμε την ορθότητα του αλγορίθμου.

- Στην τέταρτη ενότητα περιγράφουμε τον αλγόριθμο επαναγραφής που προτείνουμε για την $\mathcal{ELHI}$. Παρουσιάζουμε τις ομοιότητές του με τον αλγόριθμο για την DL-Lite, καθώς και τους νέους κανόνες που χρειάστηκε να εισάγουμε. Τέλος, αποδεικνύουμε την ορθότητα του αλγορίθμου.

- Στην πέμπτη ενότητα παρουσιάζουμε έναν αλγόριθμο επαναγραφής για την Horn-$\mathcal{SHIQ}$ που εισάγει κατάλληλες παραμέτρους στη μέθοδο της βασικής

υπέρθεσης και υποστηρίζει συζευκτικά ερωτήματα. Στη συνέχεια της ενότητας, βελτιστοποιούμε τον αλγόριθμο προσαρμόζοντας κατάλληλα τους κανόνες για την $\mathcal{ELHI}$. Επιπλέον, εισάγουμε δύο νέους κανόνες βασικής υπέρθεσης, που αφορούν τις προτάσεις με ισότητα που υποστηρίζει η γλώσσα. Τέλος, αποδεικνύουμε την ορθότητα του αλγορίθμου.

- Στην έκτη ενότητα παρουσιάζουμε επιπρόσθετους μηχανισμούς βελτιστοποίησης που αφορούν τη σειρά και τον τρόπο εφαρμογής των κανόνων της επίλυσης.

- Στην έβδομη ενότητα γίνεται η αξιολόγηση του συστήματος που υλοποιεί τους αλγορίθμους επαναγραφής που προτείνουμε για τις DL-Lite, $\mathcal{ELHI}$ και Horn-$\mathcal{SHIQ}$. Τα πειραματικά αποτελέσματα αφορούν μία σειρά από οντολογίες μεγάλης κλίμακας.

- Στην όγδοη ενότητα παρουσιάζουμε τη σχετική εργασία στο πεδίο του ενδιαφέροντός μας, και τέλος, συνοψίζουμε τη συνεισφορά της εργασίας μας.

Κεφάλαιο 2

Θεωρητικό Υπόβαθρο

Σε αυτή την ενότητα παρουσιάζουμε τις θεωρίες, τους ορισμούς, καθώς και την ορολογία που είναι απαραίτητα για την κατανόηση της εργασίας. Συγκεκριμένα, στην ενότητα 2.1 παρουσιάζουμε κάποιες βασικές έννοιες και συμβολισμούς που θα χρησιμοποιήσουμε σε όλη την εργασία, στην ενότητα 2.2 συνοψίζουμε τη θεωρία της επίλυσης και βασικές έννοιες της θεωρίας λογικής πρώτης τάξης, στην ενότητα 2.3 παρουσιάζουμε τη σύνταξη και σημασιολογία της γλώσσας περιγραφικής λογικής $\mathcal{SHIQ}$. Στην ενότητα 2.4 παρουσιάζουμε τον ορισμό του προβλήματος της επαναγραφής ερωτήματος σε οντολογίες, και περιγράφουμε την προσέγγιση του προβλήματος με τη μέθοδο της επίλυσης.

2.1 Προτάσεις λογικής πρώτης τάξης

Στις επόμενες ενότητες χρησιμοποιούμε τις έννοιες της λογικής πρώτης τάξης για τις *μεταβλητές* (*variables*), που δηλώνονται με τα γράμματα $x, y, z, \ldots$, τις *σταθερές* (*constants*), που δηλώνονται με τα $a, b, c, \ldots$, τα *συναρτησιακά σύμβολα* (*function symbols*), που δηλώνονται με τα $f, g, \ldots$. Οι όροι δηλώνονται με τα γράμματα $\sigma, \mu, \ldots$ και μπορεί να είναι μεταβλητές, σταθερές ή συναρτήσεις $f(t_1, \ldots, t_n)$, με πληθυκότητα n, όπου t_i είναι όροι. Μία λίστα από όρους της μορφής $(s_1, \ldots, s_n)$ γράφεται σε συντομία με $\vec{s}$. Επίσης, αναφερόμαστε στα *άτομα* (*atoms*), για παράδειγμα $C(x), R(x, y)$, καθώς και τη σχέση *συνεπαγωγής* (*entailement*) $\models$.

Μία αντικατάσταση *αντικατάσταση* (*substitution*) δηλώνεται με τα γράμματα σ, $\mu, \ldots$, και περιγράφεται σα μία συνάρτηση που αντιστοιχίζει μεταβλητές μίας γλώσσας σε όρους. Εάν για δύο όρους t, s ισχύει $t\theta = s\theta$, η αντικατάσταση θ ονομάζεται *ενοποιητής* (*unifier*) [10]. Η σύνθεση των αντικαταστήσεων θ, σ, συμβολίζεται με $\theta \cdot \sigma$, και για έναν όρο t ορίζεται ως $t\theta \cdot \sigma = t\theta\sigma = (t\theta)\sigma$. Ονομάζουμε *μέγιστο κοινό ενοποιητή, μκε* (*most general unifier, mgu*), έναν ενοποιητή θ τέτοιο ώστε για κάθε αντικατάσταση σ ισχύει $s\sigma = t\theta\theta'$.

Το *βάθος* (*depth*) ενός όρου, t ορίζεται ως εξής: $\mathsf{depth}(t) = 0$, αν ο t είναι σταθερά ή μεταβλητή, και $\mathsf{depth}(f(s)) = 1 + \mathsf{depth}(s)$, αν ο t περιλαμβάνει κάποια συνάρτηση. Το *βάθος ενός ατόμου* (*depth of an atom*) είναι το μέγιστο βάθος των όρων του.

Μία *θέση, p* (*position*) είναι μία πεπερασμένη σειρά ακεραίων. Χρησιμοποιούμε $t|_p$

για να δηλώσουμε τον υποόρο του t στη θέση p. Μία αντικατάσταση ενός υποόρου του t στη θέση p ορίζεται επαγωγικά ως εξής: $t[s]_\epsilon = s$ και αν $t = f(t_1, \ldots, t_n)$, τότε $t[s]_{ip} = f(t_1, \ldots, t_i[s]_p, \ldots, t_n)$.

Μία *πρόταση* (*clause*) είναι μία διάζευξη της μορφής $C_1 \vee C_2 \vee \ldots \vee C_n$ όπου C_i είναι ένα άτομο ή η άρνησή του. Για παράδειγμα, θεωρούμε την πρόταση λογικής πρώτης τάξης:

$$\forall x(\neg \textsf{Φοιτητής}(x) \vee \textsf{ΜεταπτυχιακόςΦοιτητής}(x) \vee \textsf{ΠροπτυχιακόςΦοιτητής}(x))$$

Η παραπάνω πρόταση δηλώνει ότι κάθε άτομο x μπορεί να είναι προπτυχιακός φοιτητής, είτε μεταπτυχιακός, είτε να μην είναι καθόλου φοιτητής. Η ίδια πρόταση μπορεί να γραφεί ως:

$$\textsf{ΜεταπτυχιακόςΦοιτητής}(x) \vee \textsf{ΠροπτυχιακόςΦοιτητής}(x) \leftarrow \textsf{Φοιτητής}(x)$$

όπου έχουμε παραλείψει τον καθολικό ποσοδείκτη και θεωρούμε ότι η μεταβλητή x είναι καθολική. Έτσι, η παραπάνω πρόταση γίνεται κατανοητή ως εξής: εάν κάποιος είναι φοιτητής τότε είναι είτε μεταπτυχιακός είτε προπτυχιακός φοιτητής.

Μία *Horn-πρόταση* (*Horn clause*) $\mathcal{C}$, είναι μία διάζευξη ατόμων με ένα το πολύ θετικό άτομο, της μορφής $H \vee \neg B_1 \vee \ldots \vee \neg B_n$, που γράφεται επίσης ως $H \leftarrow B_1 \wedge \ldots \wedge B_n$. Το H ονομάζεται *κεφαλή* (*head*) της πρότασης, το σύνολο $\{B_1, \ldots, B_n\}$ ονομάζεται *σώμα* (*body*), και δηλώνεται με $\textsf{body}(\mathcal{C})$. Για παράδειγμα, θεωρούμε την πρόταση:

$$\forall x(\textsf{Φοιτητής}(x) \vee \neg \textsf{ΜεταπτυχιακόςΦοιτητής}(x)) \text{ ή αλλιώς}$$

$$\forall x(\textsf{Φοιτητής}(x) \leftarrow \textsf{ΜεταπτυχιακόςΦοιτητής}(x))$$

όπως και στο προηγούμενο παράδειγμα παραλείπουμε τον καθολικό ποσοδείκτη και θεωρούμε ότι η μεταβλητή της είναι καθολική:

$$\textsf{Φοιτητής}(x) \leftarrow \textsf{ΜεταπτυχιακόςΦοιτητής}(x)$$

Έστω η παρακάτω Horn-πρόταση που διαθέτει υπαρξιακό τελεστή:

$$\forall x(\neg \textsf{Φοιτητής}(x) \vee \exists y.\textsf{παρακολουθείΜάθημα}(x, y)) \text{ ή}$$

$$\exists y.\textsf{παρακολουθείΜάθημα}(x, y) \leftarrow \textsf{Φοιτητής}(x)$$

Δηλώνει ότι κάθε φοιτητής x παρακολουθεί κάποιο μάθημα y. Είναι δυνατό να αφαιρέσουμε τον υπαρξιακό ποσοδείκτη, εαν αντικαταστήσουμε τη μεταβλητή που τον συνοδεύει από μία *skolem συνάρτηση* (*skolem function*) [9].

$$\textsf{παρακολουθείΜάθημα}(x, f(x)) \leftarrow \textsf{Φοιτητής}(x)$$

Ένας *κανόνας datalog* (*datalog rule*) r, είναι μια Horn πρόταση *χωρίς συναρτήσεις* όπου οι μεταβλητές της κεφαλής παρουσιάζονται και στο σώμα. Ένα πρόγραμμα datalog (*datalog program*), $\mathcal{P}$, είναι ένα πεπερασμένο σύνολο από datalog κανόνες.

Μία πρόταση $\mathcal{C}_1$ *υπάγει* (*subsumes*) μία $\mathcal{C}_2$, αν υπάρχει μία αντικατάσταση θ τέτοια ώστε $\mathcal{C}_1\theta = \mathcal{C}_2$ και κάθε άτομο στο σώμα της $\mathcal{C}_1\theta$ εμφανίζεται επίσης στο σώμα

της $\mathcal{C}_2$, $\mathsf{body}(\mathcal{C}_1\theta) \subseteq \mathsf{body}(\mathcal{C}_2)$. Για παράδειγμα, η πρόταση $\mathsf{ΥποψήφιοςΔιδάκτορας(x)} \leftarrow \mathsf{ΜεταπτυχιακόςΦοιτητής(x)} \wedge \mathsf{κάνειΔιαδακτορικό(x, y)}$ υπάγει την $\mathsf{ΥποψήφιοςΔιδάκτορας(x)} \leftarrow \mathsf{Φοιτητής(x)} \wedge \mathsf{ΜεταπτυχιακόςΦοιτητής(x)} \wedge \mathsf{κάνειΔιαδακτορικό(x, y)}$.

Ονομάζουμε *βασική πρόταση* (*ground clause*) μία πρόταση που περιλαμβάνει μόνο σταθερές, για παράδειγμα η πρόταση $\mathsf{παρακολουθείΜάθημα(Άννα, ΤεχνητήΝοημοσύνη)} \leftarrow \mathsf{Φοιτητής(Άννα)}$. Χρησιμοποιούμε τον όρο $\mathsf{var}(A)$ για να δηλώσουμε το σύνολο των μεταβλητών ενός ατόμου A, ενώ ο όρος var μπορεί να επεκταθεί για τις προτάσεις. Μία μεταβλητή που εμφανίζεται μόνο μία φορά στο σώμα μιας πρότασης $\mathcal{C}$ ονομάζεται *μη-δεσμευμένη μεταβλητή* (*unbound variable*) και συμβολίζεται με $\mathsf{unbound}(\mathcal{C})$. Για παράδειγμα, η y είναι μη-δεσμευμένη μεταβλητή στην πρόταση $\mathsf{Φοιτητής}(x) \leftarrow \mathsf{παρακολουθείΜάθημα}(x, y)$. Η μεταβλητή που εμφανίζεται τουλάχιστον δύο φορές στο σώμα της πρότασης $\mathcal{C}$ και δεν εμφανίζεται στην κεφαλή, εξαιρείται η περίπτωση που η κεφαλή της $\mathcal{C}$ περιλαμβάνει το κατηγόρημα ισότητας, ονομάζεται *δεσμευμένη μεταβλητή* (*bound variable*), και συμβολίζεται με $\mathsf{bound}(\mathcal{C})$. Εναλλακτικά, η δεσμευμένη μεταβλητή εμφανίζεται τουλάχιστον δύο φορές στο σώμα και δεν είναι μεταβλητή απάντησης, όπως θα δούμε παρακάτω.

Σημειώνουμε ότι στα παρακάτω χρησιμοποιούμε το συμβολισμό $[C(x)]$ για να δηλώσουμε ότι η παρουσία του ατόμου $C(x)$ είναι προαιρετική. Για παράδειγμα, η πρόταση $A(x) \leftarrow B(x) \wedge [C(x)]$ δηλώνει είτε την $A(x) \leftarrow B(x) \wedge C(x)$ είτε την $A(x) \leftarrow B(x)$. Επίσης, συχνά γράφουμε σε συντομία μία σύζευξη της μορφής $C_1(s), \wedge \ldots \wedge C_n(s)$ με $\mathbf{C}(s)$.

2.2 Επίλυση

Η *επίλυση* (*resolution*) αποτελεί μία από τις σημαντικότερες τεχνικές αυτόματης απόδειξης θεωρημάτων (automated theorem proving)[13].

Ο βασικός *κανόνας συμπερασμού* (*inference rule*) της επίλυσης είναι μία σχέση που συνήθως ορίζεται ως εξής:

$$\frac{\mathcal{C} \quad \mathcal{C}_1 \ \ldots \ \mathcal{C}_n}{\mathcal{C}'}$$

όπου η πρόταση $\mathcal{C}$ ονομάζεται *κύρια προκείμενη πρόταση* (*main premise*), οι $\mathcal{C}_1, \ldots, \mathcal{C}_n$ *δευτερεύσουσες προκείμενες προτάσεις* (*side premises*) και η πρόταση $\mathcal{C}'$ ονομάζεται *συμπέρασμα* (*resolvent*). Αν υποθέσουμε ότι έχουμε δύο προτάσεις $\mathcal{C}_1, \mathcal{C}_2$ της μορφής $A \leftarrow \mathbf{B} \wedge D$, $D' \leftarrow \mathbf{C}$ τότε ο κανόνας της *δυαδικής επίλυσης* (*binary resolution*) παράγει την πρόταση $(A \leftarrow \mathbf{B} \wedge \mathbf{C})\theta$, όπου θ είναι ένας ενοποιητής, $D\theta = D'\theta$.

Η *παραγωγή με βάση την επίλυση* (*resolution derivation*) μίας πρότασης $\mathcal{C}$ από ένα σύνολο προτάσεων Σ, είναι μία ακολουθία $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_n$, όπου η πρόταση $\mathcal{C}_n$ είναι ίδια με τη $\mathcal{C}$, και κάθε $\mathcal{C}_i$ είτε ανήκει στο Σ, είτε είναι συμπέρασμα άλλων προτάσεων που ανήκουν στην ακολουθία. Η παραγωγή με βάση την επίλυση, της πρότασης $\mathcal{C}$ από το αρχικό σύνολο προτάσεων Σ συμβολίζεται με $\Sigma \vdash \mathcal{C}$. Η παραγωγή μίας πρότασης με βάση ένα σύνολο κανόνων επίλυσης, ή αλλιώς ένα *λογισμό* (*calculus*), $\mathcal{I}$, συμβολίζεται με $\Sigma \vdash^{\mathcal{I}} \mathcal{C}$. Γράφουμε $\Sigma \vdash_i \mathcal{C}$ για να δηλώσουμε ότι το βάθος του *δένδρου παραγωγής* (*derivation tree*) [39] που κατασκευάστηκε για την $\mathcal{C}$ από το Σ είναι μικρότερο,

ή ίσο του i. Ένα σύνολο από προτάσεις είναι *κορεσμένο* (*saturated*) με βάση το σύστημα συμπερασμού $\mathcal{I}$, αν το συμπέρασμα οποιουδήποτε κανόνα του $\mathcal{I}$ από το Σ είναι στοιχείο του Σ.

Ας θεωρήσουμε ένα σύνολο προτάσεων Σ και μία πρόταση $\mathcal{C}$ που παράγεται με βάση την επίλυση από προτάσεις στο Σ, δηλαδή $\Sigma \vdash \mathcal{C}$. Εξετάζουμε εαν ισχύει η συνεπαγωγή $\Sigma \models \mathcal{C}$. Διευκρινίζουμε ότι το Σ συνεπάγεται την $\mathcal{C}$ εαν το σύνολο προτάσεων $\Sigma \cup \neg\mathcal{C}$ είναι μη ικανοποιήσιμο, δηλαδή δε διαθέτει μοντέλο. Γνωρίζουμε ότι πρόταση $\mathcal{C}$ παράγεται εφαρμόζοντας τον κανόνα της επίλυσης σε άλλες προτάσεις που ανήκουν στο Σ, ή που παράγονται από τον κανόνα της επίλυσης. Επομένως, όπως γίνεται κατανοητό, αν $\Sigma \vdash \mathcal{C}$ τότε $\Sigma \models \mathcal{C}$. Το αντίστροφο όμως, δεν ισχύει. Πιο συγκεκριμένα, ένα σύνολο Σ μπορεί να συνεπάγεται μία πρόταση $\mathcal{C}$, χωρίς η πρόταση να είναι δυνατό να παραχθεί μέσω της επίλυσης. Για παράδειγμα, έστω ότι $\Sigma = \{B\}$, ισχύει ότι $\Sigma \models A \vee \neg A$, ενώ όπως είναι προφανές, η πρόταση $A \vee \neg A$ δεν είναι δυνατό να κατασκευαστεί από το Σ με επίλυση, $\Sigma \nvdash A \vee \neg A$. Συμπεραίνουμε ότι η μέθοδος της επίλυσης είναι ορθή αλλά όχι πλήρης.

Σε αυτό το σημείο, εξετάζουμε την περίπτωση που ένα σύνολο προτάσεων συνεπάγεται την κενή πρόταση. Συγκεκριμένα, εάν ένα σύνολο Σ συνεπάγεται την κενή πρόταση $\Sigma \models []$, τότε η κενή πρόταση είναι δυνατό να παραχθεί εφαρμόζοντας τους κανόνες της επίλυσης, $\Sigma \vdash []$. Από την ορθότητα της μεθόδου, συμπεραίνουμε ότι ισχύει και το αντίστροφο. Λέμε ότι η μέθοδος της επίλυσης είναι *πλήρης ως προς την άρνηση* (*refutation complete*). Έτσι λοιπόν, εαν ένα σύνολο προτάσεων στη λογική πρώτης τάξης είναι μη ικανοποιήσιμο, τότε με την μέθοδο της επίλυσης μπορούμε να συμπεράνουμε την κενή πρόταση. Την ίδια στιγμή όμως, ενδέχεται η διαδικασία εφαρμογής των κανόνων της επίλυσης να μην τερματίζει.

Για τους σκοπούς της εργασίας περιοριζόμαστε σε λογισμούς που αφορούν ένα υποσύνολο της λογικής πρώτης τάξης (την περιγραφική λογική Horn-$\mathcal{SHIQ}$). Από τα παραπάνω, προκύπτει άμεσα η ορθότητα των λογισμών επαναγραφής που εφαρμόζουν την επίλυση, ενώ, είναι απαραίτητη η απόδειξη για την πληρότητα και τον τερματισμό των αλγορίθμων.

Μία παραγωγή επίλυσης με ιδιαίτερο ενδιαφέρον για την εργασία μας είναι η *SLD*. Η *παραγωγή επίλυσης SLD* (*SLD resolution derivation*) μίας πρότασης $\mathcal{C}$ από ένα σύνολο προτάσεων Σ είναι μία ακολουθία από προτάσεις $\mathcal{C}_1, \ldots, \mathcal{C}_n$ τέτοια ώστε $\mathcal{C}_1 \in \Sigma$, $\mathcal{C}_n = \mathcal{C}$ και $\mathcal{C}_{i+1}$ είναι συμπέρασμα της $\mathcal{C}_i$ και κάποιας πρότασης από το Σ. Χρησιμοποιούμε το συμβολισμό $\mathcal{I}_{\mathsf{SLD}}$ για να δηλώσουμε το λογισμό εκείνο που μπορεί να παράγει μία ακολουθία προτάσεων με την παραπάνω ιδιότητα.

2.2.1 Επίλυση σε προτάσεις με ισότητα

Η μέθοδος της επίλυσης επεκτείνεται με επιπλέον κανόνες στην περίπτωση μίας λογικής που υποστηρίζει την ισότητα. Η *ισότητα* (*equality*) είναι μία έκφραση της μορφής $s \approx t$, όπου s, t υποδηλώνουν κάποιο όρο ή κατηγόρημα. Η ισότητα διεθέτει συμμετρική, αντιμεταθετική και μεταβατική ιδιότητα. Η άρνηση μιας ισότητας $\neg(s \approx t)$ γράφεται ως $s \not\approx t$.

Στην περίπτωση των προτάσεων με ισότητα η μέθοδος της επίλυσης πρέπει να

προσαρμοστεί κατάλληλα, ώστε να λαμβάνονται υπόψη οι ιδιότητες του κατηγορήματος $\approx$. Για παράδειγμα, για τη συνεπαγωγή της πρότασης $t \approx a$ από ένα σύνολο $\{s \approx a, s \approx t\}$, είναι απαραίτητο να ληφθούν υπόψη η συμμετρική και μεταβατική ιδιότητα του κατηγορήματος της ισότητας. Η βασική τεχνική για την απόδειξη θεωρημάτων σε γλώσσες που περιλαμβάνουν προτάσεις με ισότητα, είναι η τεχνική της *υπέρθεσης* (*superposition*) [14, 101]. Σημαντικές βελτιστοποιήσεις της υπέρθεσης οδήγησαν στη μέθοδο της *βασικής υπέρθεσης* (*basic superposition*) [11, 12], η οποία θα μας απασχολήσει στη συγκεκριμένη εργασία.

Για τους σκοπούς της θεωρίας των αποδείξεων σε λογικές με ισότητα, ένα κατηγόρημα $D(s)$ γράφεται ως $D(s) \approx \top$. Επιπλέον, μία πρόταση $C_1 \vee \ldots C_n$ γράφεται ως $(C'_1 \vee C'_2 \vee \ldots \vee C'_n) \cdot \sigma$ με $C'_i\sigma = C_i$ όπου $C_1 \vee \ldots C_n$ είναι ο *σκελετός της πρότασης* (*skeleton of a closure*) και σ είναι μία αντικατάσταση που αναπαριστά τα αποτελέσματα προηγούμενων αντικαταστήσεων. Για μία πρόταση $(A_1 \vee \ldots \vee A_n) \cdot \sigma$ θεωρούμε ότι η αντικατάσταση σ δεν αντιστοιχίζει μεταβλητές σε μεταβλητές και ότι το πεδίο ορισμού της ανήκει στο σύνολο των μεταβλητών της $A_1 \vee \ldots \vee A_n$. Για παράδειγμα, μία πρόταση $A(f(v)) \vee \neg C(v)$ που έχει προκύψει από επίλυση της $A(x) \vee \neg B(x)$ με την $B(f(v)) \vee \neg C(v)$ γράφεται και ως $(A(x) \vee C(v)) \cdot \sigma$ με $\sigma = \{x \mapsto f(v)\}$.

Η μέθοδος της βασικής υπέρθεσης προσδιορίζεται από δύο παραμέτρους, μία *διάταξη αναγωγής* (*reduction ordering*), *πλήρη* (*total*) για τους *βασικούς όρους* (*ground terms*) και μία *συνάρτηση επιλογής* (*selection function*), που επιλέγει αρνητικά άτομα σε μία πρόταση $A \vee \neg C_1 \vee \ldots \vee \neg C_n$, ή αλλιώς άτομα στο σώμα μίας πρότασης $A \leftarrow C_1 \vee \ldots \vee C_n$. Μία πλήρης διάταξη βασικών όρων, $\succ$, επεκτείνεται για τα άτομα με τον εξής τρόπο: σε ένα άτομο $s \bowtie t$ αντιστοιχίζεται μία τριάδα $c_B = (max(s,t), p, min(s,t))$, όπου max, min επιστρέφουν το μέγιστο και ελάχιστο των όρων s, t, και $p = 1$, αν το $\bowtie$ υποδηλώνει $\not\approx$, ενώ $p = 0$, αν το $\bowtie$ υποδηλώνει $\approx$. Οι τριάδες c_B συγκρίνονται λεξικογραφικά με $1 \succ 0$. Θεωρούμε ότι $f \succ c \succ P \succ \top$, όπου $\top$ είναι η καθολική έννοια, P είναι ένα κατηγόρημα, c σταθερά και f μία συνάρτηση. Επιλέον, $a \not\approx b \succ a \approx b$. Λέμε ότι ένα άτομο B_k είναι *(αυστηρά)μέγιστο* (*(strictly) maximal*) σε μία πρόταση όταν δεν υπάρχει άλλο άτομο B_l τέτοιο ώστε $B_l(\succeq) \succ B_k$. Ένα άτομο είναι *κατάλληλο για επίλυση* (*eligible for resolution*) σε μία πρόταση εαν η επιλέγεται από τη συνάρτηση επιλογής, ή δεν υπάρχουν άτομα επιλεγμένα και είναι αυστηρά μέγιστο στην πρόταση. Ένα άτομο είναι *(αυστηρά) κατάλληλο για υπέρθεση* (*(strictly) eligible for superposition*) σε μία πρόταση, εαν δεν υπάρχουν επιλεγμένα άτομα στην πρόταση και είναι (αυστηρά) μέγιστο στην πρόταση. Οι κανόνες της βασικής υπέρθεσης που εφαρμόζονται στην λογική που θα μας απασχολήσει σε αυτή την εργασία παρουσιάζονται στον Πίνακα 2.1.

2.3 Γλώσσες ΠΛ Horn-$\mathcal{SHIQ}$

Έστω $\mathbf{C}$, $\mathbf{R}$, and $\mathbf{I}$ μετρήσιμα, ανά ζεύγη ξένα σύνολα ατομικών εννοιών, ατομικών ρόλων και ατόμων. Ένας $\mathcal{SHIQ}$-ρόλος είναι είτε ένας ρόλος $R \in \mathbf{R}$, είτε ο αντίστροφός του R^-. Για $A \in \mathbf{C}$, $\top, \bot$ δύο έννοιες που ονομάζουμε καθολική και κενή έννοια, αντίστοιχα, $R \in \mathbf{R}$, και n ένα θετικό ακέραιο, το σύνολο των $\mathcal{SHIQ}$-

Πίνακας 2.1: Κανόνες βασικής θετικής υπέρθεσης για τη Horn-$\mathcal{SHIQ}$

θετική υπέρθεση:

$$\frac{(w \approx v \leftarrow \bigwedge_i D_i(\vec{s_i}))\rho \quad (t_1 \approx t_2 \leftarrow \bigwedge_j D_j(\vec{t_j}))\rho}{(w[t_2]_p \approx v \leftarrow D_i(\vec{s_i}))\theta}$$

όπου i) $\sigma = \mathsf{mgu}(w|_p\rho, t_1\rho), \theta = \rho\sigma$ ii) $v\theta \not\succeq w\theta$ και $t_2\theta \not\succeq t_1\theta$ iii) το $(t_1 \approx t_2)\theta$ είναι αυστηρά κατάλληλο για υπέρθεση στο $(t_1 \approx t_2 \leftarrow \bigwedge_j D_j(\vec{t_j}))\theta$ iv) το $(w \approx v)\theta$ είναι αυστηρά κατάλληλο για υπέρθεση στο $(w \approx v \leftarrow \bigwedge_i D_i(\vec{s_i}))\theta$ v) $t_1\theta \approx t_2\theta \not\succeq w\theta \approx v\theta$ vi) το $w|_p$ δεν είναι μεταβλητή

ταξινομημένη υπερεπίλυση:

$$\frac{N \quad E_1, E_2, \ldots E_n}{(C_1 \vee C_2 \vee \ldots_n \vee D)\theta}$$

όπου i) E_i είναι προτάσεις της μορφής $(C_i \vee A_i)\rho$ για $1 \leq i \leq n$ ii) N είναι μία πρόταση της μορφής $(D \vee \neg B_1 \vee \ldots \vee \neg B_n)\rho$ iii) σ είναι ο μκε τ.ω. $A_i\theta = B_i\theta$ για $1 \leq i \leq n$ και $\theta = \rho\sigma$ iv) κάθε άτομο $A_i\theta$ είναι αυστηρά κατάλληλο για υπέρθεση στο E_i iv) είτε επιλέγονται όλα τα $\neg B_i\theta$, είτε δεν επιλέγεται κανένα, και το $\neg B_i\theta$ είναι μέγιστο στο $D\theta$.

εννοιών ορίζεται επαγωγικά με την ακόλουθη σύνταξη:

$$B, C := \top \mid \bot \mid A \mid \neg B \mid B \sqcup C \mid B \sqcap C \mid \exists R.C \mid \forall R.C \mid \leq nR.C \mid \geq nR.C$$

Μια $\mathcal{SHIQ}$ *οντολογία*, ή *σώμα ορισμού*, TBox $\mathcal{T}$, είναι ένα σύνολο που περιλαμβάνει αξιώματα υπαγωγής εννοιών της μορφής $C \sqsubseteq D$, αξιώματα υπαγωγής ρόλων $R \sqsubseteq S$, και αξιώματα μεταβατικότητας ρόλων, της μορφής $\mathsf{Tra}(R)$. Ονομάζουμε έναν ρόλο R *απλό* όταν δεν είναι μεταβατικός $\mathsf{Tra}(R) \in \mathcal{T}$ και δε διαθέτει κάποιο μεταβατικό υπορόλο, δηλαδή κάποιο S τέτοιο ώστε $\mathsf{Tra}(S) \in \mathcal{T}$ και $S \sqsubseteq^* R$, όπου $\sqsubseteq^*$ είναι το μεταβατικό κλείσιμο του $\{(S, R) \mid S \sqsubseteq R \in \mathcal{T}$ ή $S^- \sqsubseteq R^- \in \mathcal{T}\}$. Προκειμένου να διασφαλιστεί η αποφανσιμότητα οι ρόλοι που εμφανίζονται σε έννοιες της μορφής $\geq nR.C$ και $\leq nR.C$ είναι απλοί.

Μία *Horn-$\mathcal{SHIQ}$ οντολογία* είναι μία $\mathcal{SHIQ}$ οντολογία ισχύουν τα εξής:

- οι έννοιες $B \sqcup C$, $\leq nR.C, n > 1$ δεν εμφανίζονται στο δεξί σκέλος των αξιωμάτων υπαγωγής εννοιών
- οι έννοιες $\neg B$, $\leq nR.C$ με $n > 1$, $\geq nR.C, \forall R.C$ δεν εμφανίζονται στο αριστερό σκέλος των αξιωμάτων υπαγωγής εννοιών

Η γλώσσα περιγραφικής λογικής $\mathcal{ELHI}$ προκύπτει αφαιρώντας από την Horn-$\mathcal{SHIQ}$ έννοιες της μορφής $A \sqcup B, \neg B, \leq nS.B, \geq mS.B, \forall S.B, \bot$ [6]. Η γλώσσα περιγραφικής λογικής DL-Lite$_{R,\sqcap}$ προκύπτει από την $\mathcal{ELHI}$ αφαιρώντας αξιώματα της μορφής $\exists S.A \sqsubseteq B$ με $A \neq \top$ [4]. Στα επόμενα, αναφερόμαστε σε αυτή γλώσσα με το όνομα DL-Lite για λόγους απλότητας.

Λέμε ότι μία έννοια C είναι *απλή* (*simple concept*) όταν είναι της μορφής $\bot$, $A, \exists R.A, \forall R.A, \leq 1S.A$, όπου A είναι ατομική έννοια και S απλός ρόλος. Χωρίς βλάβη της γενικότητας θεωρούμε ότι οι Horn-$\mathcal{SHIQ}$ οντολογίες είναι *κανονικοποιημένες* (*normalized ontology*) [76], δηλαδή περιλαμβάνουν αξιώματα της μορφής $\sqcap A_i \sqsubseteq C$, $\mathsf{Tra}(R), R_1 \sqsubseteq R_2$, όπου $A_{(i)}, R_{(i)}$ είναι ατομικές έννοιες και ρόλοι, C είναι μία απλή έννοια, ενώ θεωρούμε ότι η κενή σύζευξη αντιστοιχεί στην καθολική έννοια $\top$. Επιπλέον, υιοθετούμε την παρακάτω τεχνική για την απομάκρυνση των αξιωμάτων

μεταβατικότητας [76]: ένα αξίωμα μεταβατικότητας $\mathsf{Tra}(R)$ μπορεί να απομακρυνθεί από την οντολογία εισάγοντας για κάθε αξίωμα $A \sqsubseteq \forall S.B$ με $R \sqsubseteq S$, τα αξιώματα $A \sqsubseteq \forall R.B'$, $B' \sqsubseteq B$, $B' \sqsubseteq \forall R.B'$, όπου B' είναι μία νέα έννοια.

Ένα Horn-$\mathcal{SHIQ}$ σώμα ισχυρισμών ABox, $\mathcal{A}$ είναι ένα σύνολο από ισχυρισμούς εννοιών και ρόλων της μορφής $a : C$, $(a, b) : R$. Ονομάζουμε Horn-$\mathcal{SHIQ}$ *βάση γνώσης* (*knowledge base*) ένα ζεύγος $\mathcal{K} = \{\mathcal{T}, \mathcal{A}\}$, όπου $\mathcal{T}$ είναι ένα Horn-$\mathcal{SHIQ}$ TBox και $\mathcal{A}$ είναι ένα Horn-$\mathcal{SHIQ}$ ABox.

Πίνακας 2.2: $\mathcal{SHIQ}$-έννοιες, ρόλοι, και αξιώματα

$\mathcal{SHIQ}$-ρόλοι	
R^-	αντίστροφος ρόλος
$\mathcal{SHIQ}$-έννοιες	
$\top$	καθολική έννοια
$\bot$	κενή έννοια
$C_1 \sqcap C_2$	τομή εννοιών
$C_1 \sqcup C_2$	ένωση εννοιών
$\exists P.C$	προσοντούχος υπαρξιακός περιορισμός
$\forall P.C$	καθολικός περιορισμός
$\leq nP.C$	ελάχιστος προσοντούχος περιορισμός πληθυκότητας
$\geq nP.C$	μέγιστος προσοντούχος περιορισμός πληθυκότητας
αξιώματα	
$C_1 \sqsubseteq C_2$	υπαγωγή εννοιών
$P_1 \sqsubseteq P_2$	υπαγωγή ρόλων
$\mathsf{Tra}(P)$	αξίωμα μεταβατικότητας
$a : C$	ισχυρισμός έννοιας
$(a, b) : P$	ισχυρισμός ρόλου

Η σημασιολογία της $\mathcal{SHIQ}$ δίνεται στα πλαίσια της διερμηνείας επί ενός πεδίου Δ^I. Μία διερμηνεία αντιστοιχίζει στιγμιότυπα σε αντικείμενα του πεδίου, έννοιες σε υποσύνολα του πεδίου και ρόλους σε ζεύγη συνόλων του πεδίου ($\Delta^\mathcal{I} \times \Delta^\mathcal{I}$). Η καθολική έννοια $\top$ αντιστοιχίζεται στο $\Delta^\mathcal{I}$, η κενή έννοια $\bot$ αντιστοιχίζεται στο κενό σύνολο, ενώ οι υπόλοιπες $\mathcal{SHIQ}$-έννοιες και $\mathcal{SHIQ}$-ρόλοι αντιστοιχίζονται σε υποσύνολα του $\Delta^\mathcal{I}$ και $\Delta^\mathcal{I} \times \Delta^\mathcal{I}$, αντίστοιχα. Για παράδειγμα, μία διερμηνεία $\mathcal{I}$ αντιστοιχίζει τη σύνθετη έννοια $C \sqcap B$ στην τομή των συνόλων στα οποία αντιστοιχίζονται οι C και B με βάση το $\mathcal{I}$—δηλαδή, $(C \sqcap B)^\mathcal{I} = C^\mathcal{I} \cap B^\mathcal{I}$. Επίσης, λέμε ότι μία διερμηνεία $\mathcal{I}$ *ικανοποιεί* ένα αξίωμα $C \sqsubseteq B$ αν η $\mathcal{I}$ αντιστοιχίζει την C σε ένα υποσύνολο του συνόλου στο οποίο αντιστοιχίζεται η B, δηλαδή αν $C^\mathcal{I} \subseteq B^\mathcal{I}$. Λέμε διερμηνεία $\mathcal{I}$ ικανοποιεί τον ισχυρισμό $R(a, b)$ αν $(a^\mathcal{I}, b^\mathcal{I}) \in R^\mathcal{I}$ και τον $A(b)$ αν $B^\mathcal{I} \in A^\mathcal{I}$. Εάν υπάρχει διερμηνεία, $\mathcal{I}$ που ικανοποιεί όλα τα αξιώματα και τους ισχυρισμούς μίας βάσης γνώσης $\mathcal{K} = \{\mathcal{T}, \mathcal{A}\}$ τότε η $\mathcal{K}$ ονομάζεται ικανοποιήσιμη *ικανοποιήσιμη* (*satisfiable*) (ή *συνεπής* (*consistent*)) και η $\mathcal{I}$ είναι *μοντέλο* (*model*) της $\mathcal{K}$, διαφορετικά λέμε ότι η βάση γνώσης είναι *μη-ικανοποιήσιμη* (*unsatisfiable*) (ή *ασυνεπής* (*inconsistent*)).

Επιπλέον, η σημασιολογία της $\mathcal{SHIQ}$ μπορεί να προκύψει από τη γνωστή με-

Πίνακας 2.3: Μετάφραση Horn-$\mathcal{SHIQ}$ αξιωμάτων σε προτάσεις

Αξίωμα	Πρόταση
$\top \sqsubseteq A(x)$	$A(x)$
$B \sqsubseteq A$	$A(x) \leftarrow B(x)$
$B_1 \sqcap B_2 \sqsubseteq A$	$A(x) \leftarrow B_1(x) \wedge B_2(x)$
$A \sqsubseteq \exists R.B$	$R(x, f(x)) \leftarrow A(x),\ B(f(x)) \leftarrow A(x)$
$A \sqsubseteq \exists R^-.B$	$R(f(x), x) \leftarrow A(x),\ B(f(x)) \leftarrow A(x)$
$A \sqsubseteq \forall R.B$	$B(x) \leftarrow R(y, x) \wedge A(y)$
$A \sqsubseteq \forall R^-.B$	$B(x) \leftarrow R(x, y) \wedge A(y)$
$A \sqsubseteq \leq 1R.B$	$y \approx z \leftarrow A(x) \wedge R(x, y) \wedge B(y) \wedge R(x, z) \wedge B(z)$
$A \sqsubseteq \leq 1R^-.B$	$y \approx z \leftarrow A(x) \wedge R(y, x) \wedge R(z, x) \wedge B(y) \wedge B(z)$
$P \sqsubseteq R$	$R(x, y) \leftarrow P(x, y)$
$P \sqsubseteq R^-$	$R(x, y) \leftarrow P(y, x)$

τάφραση στη λογική πρώτης τάξης [9]. Ο πίνακας 2.3 παρουσιάζει τη μετάφραση μιας κανονικοποιημένης Horn-$\mathcal{SHIQ}$ οντολογίας χωρίς αξιώματα μεταβατικότητας σε λογική πρώτης τάξης. Όπως είναι φανερό, οι υπαρξιακές μεταβλητές της λογικής πρώτης τάξης έχουν αντικατασταθεί από μία skolem συνάρτηση. Επομένως, κάθε αξίωμα με υπαρξιακό τελεστή στο δεξί σκέλος, $A \sqsubseteq \exists R.B$ σχετίζεται μονοσήμαντα με ένα συναρτησιακό σύμβολο, f. Τα A, B αναπαριστούν ατομικές έννοιες. Επιπλέον, στην εργασία μας θεωρούμε την απάντηση ερωτημάτων για συνεπείς βάσεις γνώσης, δεν έχουμε συμπεριλάβει αξιώματα της μορφής $A \sqsubseteq \neg B, A \sqsubseteq \bot$ που μπορούν να οδηγήσουν σε ασυνέπειες, για παράδειγμα στην περίπτωση ενός ABox $\mathcal{A} = \{a : A, a : B\}$ [21, 24, 22, 87].

Στα επόμενα θεωρούμε ότι ένα Horn-$\mathcal{SHIQ}$ ABox $\mathcal{A}$ περιλαμβάνει ισχυρισμούς της μορφής $A(a), R(a, b)$, όπου A είναι ατομική έννοια.

2.4 Λογισμοί επαναγραφής

Η επαναγραφή ερωτήματος είναι μία επιφανής τεχνική για την απάντηση ερωτημάτων πάνω σε οντολογίες. Στα επόμενα παρουσιάζουμε τυπικά την έννοια του συζευκτικού ερωτήματος καθώς και της επαναγραφής συζευκτικού ερωτήματος με βάση κάποια βάση γνώσης.

Ένα *συζευκτικό ερώτημα, ΣΕ* (*conjunctive query*) $\mathcal{Q}$, είναι ένας κανόνας datalog με το *κατηγόρημα ερωτήματος* (*query predicate*) Q στην κεφαλη που δεν εμφανίζεται πουθενά στο σώμα. Ένα *Σύνολο από Συζευκτικά Ερωτήματα, ΣΣΕ* (*Union of Conjuctive Queries, UCQ*), είναι ένα είναι ένα σύνολο συζευκτικών ερωτημάτων με το ίδιο κατηγόρημα ερωτήματος. Οι μεταβλητές της κεφαλής ενός ΣΕ $\mathcal{Q}$ ονομάζονται *μεταβλητές απάντησης* (*answer variables*) και συμβολίζονται με $\mathsf{avar}(\mathcal{Q})$. Ονομάζουμε *ορισμένη απάντηση* (*certain answer*) ενός ερωτήματος $\mathcal{Q}$, με κατηγόρημα ερωτήματος Q, με βάση ένα σώμα ορολογίας TBox, $\mathcal{T}$ και ένα σώμα ισχυρισμών ABox, $\mathcal{A}$, μία πλειάδα σταθερών $\vec{a}$ με πληθυκότητα που συμφωνεί με αυτή του Q και $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\} \models Q(\vec{a})$, συβολίζουμε το σύνολο των απαντήσεων με $\mathsf{cert}(\mathcal{Q}, \mathcal{T}, \mathcal{A})$.

Η *επαναγραφή* (*rewriting*) ενός ερωτήματος $\mathcal{Q}$ με βάση ένα σώμα ορολογίας $\mathcal{T}$ μπορεί να περιγραφεί σαν ένα σύνολο προτάσεων (συνήθως ένα πρόγραμμα datalog ή ένα ΣΣΕ) που συγκεντρώνει όλη τη σχετική πληροφορία από το $\mathcal{T}$ για την απάντηση του $\mathcal{Q}$ πάνω από οποιοδήποτε σώμα ισχυρισμών $\mathcal{A}$ [33, 110, 116].

Ορισμός 2.4.1. Έστω ένα ΣΕ $\mathcal{Q}$ με κατηγόρημα ερωτήματος Q και έστω ένα σώμα ορολογίας $\mathcal{T}$. Μία *datalog επαναγραφή* (ή απλά *επαναγραφή*) $\mathcal{R}$ του ΣΕ $\mathcal{Q}$ με βάση το $\mathcal{T}$ είναι ένα datalog πρόγραμμα του οποίου οι κανόνες μπορούν να διαμεριστούν σε δύο ξένα σύνολα $\mathcal{R}_D$ και $\mathcal{R}_Q$, τέτοια ώστε το $\mathcal{R}_D$ δεν αναφέρει το Q, το $\mathcal{R}_Q$ είναι ένα ΣΣΕ με κατηγόρημα ερωτήματος Q, και για κάθε $\mathcal{A}$ συνεπές με βάση το $\mathcal{T}$ έχουμε:

$$\mathsf{cert}(\mathcal{Q}, \mathcal{T}, \mathcal{A}) = \mathsf{cert}(\mathcal{R}_Q, \mathcal{R}_D, \mathcal{A}).$$

Αν $\mathcal{R}_D = \emptyset$, τότε ονομάζουμε το $\mathcal{R}$ *επαναγραφή ΣΣΕ* (*UCQ rewriting*). △

Παράδειγμα 2.4.2. Θεωρούμε το ακόλουθο σώμα ορολογίας και ερώτημα:

$$\begin{aligned} \mathcal{T}_1 &= \{A \sqsubseteq \exists R.B, R \sqsubseteq S, B \sqcap E \sqsubseteq C\} \\ \mathcal{Q}_1 &= Q(x) \leftarrow S(x,y) \wedge C(y) \end{aligned}$$

Μπορεί να επιβεβαιωθεί ότι το σύνολο $\mathcal{R} = \{\mathcal{Q}_1, \mathcal{C}_1, \mathcal{C}_2\}$ όπου $\mathcal{C}_1 = S(x,y) \leftarrow R(x,y)$ και $\mathcal{C}_2 = C(x) \leftarrow B(x) \wedge E(x)$ είναι μία datalog επαναγραφή του $\mathcal{Q}_1$ με βάση το $\mathcal{T}_1$. ◇

2.4.1 Επαναγραφή με επίλυση

Όπως ήδη γνωρίζουμε, η τεχνική της επίλυσης είναι συνεπής και πλήρης ως προς την άρνηση σε υποσύνολα της λογικής πρώτης τάξης. Έτσι, προκειμένου να αποφανθούμε εαν μία πλειάδα a είναι απάντηση ενός ερωτήματος $\mathcal{Q}$, σε μία Horn-$\mathcal{SHIQ}$ βάση γνώσης $\mathcal{K}$, θεωρούμε το σύνολο που αποτελείται από τις προτάσεις της βάσης γνώσης, και την πρόταση $\bot \leftarrow Q(a)$, όπου Q είναι η κεφαλή του $\mathcal{Q}$. Έπειτα, εξετάζουμε εάν μπορούμε να παράγουμε την κενή πρόταση εφαρμόζοντας τους κανόνες της επίλυσης, δηλαδή $\mathcal{K} \models Q(a)$ ανν $\{\mathcal{K}, \bot \leftarrow Q(a)\} \vdash []$.

Οι αλγόριθμοι επαναγραφής ενός ερωτήματος $\mathcal{Q}$ με βάση ένα σώμα ορολογίας $\mathcal{T}$, και ένα σώμα ισχυρισμών, $\mathcal{A}$, που βασίζονται στην επίλυση ακολουθούν μία διαδικασία που σε γενικές γραμμές μπορεί να περιγραφεί ως εξής: η βάση γνώσης $\mathcal{K} = \{\mathcal{T}, \mathcal{A}\}$ μετατρέπεται σε ένα σύνολο προτάσεων λογικής πρώτης τάξης προκειμένου να εφαρμοστούν οι κανόνες της επίλυσης. Αποδεικνύεται ότι η εφαρμογή των κανόνων σε προτάσεις του σώματος ισχυρισμών είναι δυνατό να αναβληθεί, μέχρις ότου οι κανόνες εφαρμοστούν εξαντλητικά (διαδικασία κορεσμού) στις προτάσεις που αντιστοιχούν στο σώμα ορολογίας $\mathcal{T}$, και στο ερώτημα $\mathcal{Q}$, καταλήγοντας έτσι σε ένα νέο σύνολο προτάσεων. Από το κατασκευασθέν σύνολο προκύπτει ένα τελικό datalog πρόγραμμα, που συνιστά την επαναγραφή, και αποτιμάται λαμβάνοντας υπόψη το σώμα ισχυρισμών, $\mathcal{A}$.

Στις ενότητες που ακολουθούν περιγράφουμε δύο αλγορίθμους επαναγραφής που βασίζονται στην επίλυση. Αρχικά, παρουσιάζουμε τον αλγόριθμο για τη $\mathcal{ELHI}$ που υλοποιήθηκε στο σύστημα Requiem [110] και στη συνέχεια, τον αλγόριθμο για τη Horn-$\mathcal{SHIQ}$ που υλοποιήθηκε στο σύστημα KAON2 [70]

Πίνακας 2.4: Τύποι προτάσεων DL-Lite ή $\mathcal{ELHI}$

Τύπος	Πρόταση
1	$Q(\vec{s}) \leftarrow \bigwedge D_j(\vec{t_j})$
2.1	$R(x, f(x)) \leftarrow A(x)$
2.2	$R(f(x), x) \leftarrow A(x)$
2.3	$B(f(x)) \leftarrow \mathbf{C}(x) \wedge [\mathbf{A}(f(x))]$
3.1	$R(x, y) \leftarrow P(x, y)$
3.2	$R(x, y) \leftarrow P(y, x)$
3.3	$A(x) \leftarrow \mathbf{C}(x) \wedge [\mathbf{B}(f(x))]$
4.1	$A(x) \leftarrow R(x, y) \wedge [\mathbf{C}(y)]$
4.2	$A(x) \leftarrow R(y, x) \wedge [\mathbf{C}(y)]$

Requiem

Έστω ένα ΣΕ $\mathcal{Q}$ και ένα σώμα ορολογίας $\mathcal{T}$ σε $\mathcal{ELHI}$. Ο αλγόριθμος του Requiem για τη δεδομένη είσοδο μπορεί να περιγραφεί από τα παρακάτω βήματα:

1. *Μετατροπή σε προτάσεις* (*clausification*): Αρχικά, το σώμα ορολογίας $\mathcal{T}$ της εισόδου μεταφράζεται σε ένα σύνολο από (Horn) προτάσεις $\mathcal{T}_\mathcal{C}$ κάνοντας χρήση της γνωστής ισοδυναμίας των αξιωμάτων ΠΛ με τις προτάσεις λογικής πρώτης τάξης και αντικαθιστώντας τις υπαρξιακές μεταβλητές με συναρτήσεις skolem [9].

2. *Κορεσμός* (*saturation*): Στη συνέχεια, το σώμα ορολογίας μαζί με το ΣΕ, αφού μετατραπούν σε προτάσεις, υπόκεινται εξαντλητικά στη διαδικασία της (δυαδικής) επίλυσης με συνάρτηση ελεύθερης επιλογής [13] και παράγουν ένα νέο σύνολο από προτάσεις $\mathcal{T}_\mathcal{C}^{sat}$. Οι κανόνες επίλυσης που εφαρμόζει ο λογισμός του Requiem μπορούν να παράγουν συγκεκριμένους τύπους προτάσεων που φαίνονται στον Πίνακα 2.4.

3. *Εκτύλιξη* (*unfolding*): Εφαρμόζει τη διαδικασία εκτύλιξης στο σύνολο $\mathcal{T}_\mathcal{C}^{sat}$ ώστε να παράγει ένα πρόγραμμα datalog βέλτιστης μορφής [110]. Παρόλο που η συγκεκριμένη διαδικασία είναι προαιρετική, τη λαμβάνουμε υπόψη και στις αποδείξεις μας σα βήμα του Requiem αλγορίθμου.

4. *Τελική επεξεργασία (post-processing)*: Τελικά, το Requiem επιστρέφει στην έξοδό του όλες τις προτάσεις από το $\mathcal{T}_\mathcal{C}^{sat}$ που δεν διαθέτουν συναρτήσεις.

Χρησιμοποιούμε το συμβολισμό $\mathcal{I}_{\mathsf{REQ}}$ για να δηλώσουμε το λογισμό που εφαρμόζεται στο βήμα 2 και $\mathsf{RQR}(\mathcal{Q}, \mathcal{T})$ για να δηλώσουμε το datalog πρόγραμμα που επιστρέφεται στο βήμα 4. Στον Πίνακα 2.5 φαίνονται όλες οι αλληλεπιδράσεις (τύποι κανόνων συμπερασμού) που λαμβάνουν χώρα όταν εφαρμόζεται ο $\mathcal{I}_{\mathsf{REQ}}$ σε προτάσεις που εκφράζονται σε $\mathcal{ELHI}$. Επεξηγούμε το λογισμό $\mathcal{I}_{\mathsf{REQ}}$ στο παρακάτω παράδειγμα.

Παράδειγμα 2.4.3. Θεωρούμε το σώμα ορολογίας και το ερώτημα του παραδείγματος 2.4.2. Όπως δηλώσαμε παραπάνω το πρώτο βήμα είναι να μετατρέψουμε το

Πίνακας 2.5: Πιθανοί τύποι κανόνων συμπερασμού με βάση το $\mathcal{I}_{\mathsf{REQ}}$

3.3+2.3=2.3

$$\frac{C(x) \leftarrow A(x) \wedge \mathbf{B}(x) \quad A(f(x)) \leftarrow \mathbf{F}(x)}{C(f(x)) \leftarrow \mathbf{B}(f(x)) \wedge \mathbf{F}(x)}$$

3.3+2.3=3.3

$$\frac{E(x) \leftarrow B(f(x)) \wedge \mathbf{C}(f(x)) \wedge \mathbf{D}(x) \quad B(f(x)) \leftarrow \mathbf{G}(x)}{E(x) \leftarrow \mathbf{C}(f(x)) \wedge \mathbf{I}(x)}$$

2.3+2.3=2.3

$$\frac{E(f(x)) \leftarrow B(f(x)) \wedge \mathbf{C}(f(x)) \wedge \mathbf{D}(x) \quad B(f(x)) \leftarrow \mathbf{G}(x)}{E(f(x)) \leftarrow \mathbf{C}(f(x)) \wedge \mathbf{I}(x)}$$

4.1+2.1=3.3

$$\frac{B(x) \leftarrow P(x,y) \wedge [\mathbf{C}(y)] \quad P(x,f(x)) \leftarrow A(x)}{B(x) \leftarrow A(x) \wedge [\mathbf{C}(f(x))]}$$

4.1+2.2=2.3

$$\frac{B(x) \leftarrow P(x,y) \wedge [\mathbf{C}(y)] \quad P(f(x),x) \leftarrow A(x)}{B(f(x)) \leftarrow A(x) \wedge [\mathbf{C}(x)]}$$

3.1+2.1=2.1

$$\frac{S(x,y) \leftarrow P(x,y) \quad P(x,f(x)) \leftarrow A(x)}{S(x,f(x)) \leftarrow A(x)}$$

3.1+2.2=2.2

$$\frac{S(x,y) \leftarrow P(x,y) \quad P(f(x),x) \leftarrow A(x)}{S(f(x),x) \leftarrow A(x)}$$

1+2.3=1

$$\frac{\mathcal{Q}(\vec{s}) \leftarrow \bigwedge D_j(\vec{t_j}) \wedge R(t_1,t_2) \wedge B(t) \quad B(f(x)) \leftarrow \mathbf{C}(x)}{\mathcal{Q}(\vec{s})\sigma \leftarrow \bigwedge D_j(\vec{t_j})\sigma \wedge R(t_1,t_2)\sigma \wedge \mathbf{C}(x)\sigma}$$

1+2.1=1

$$\frac{\mathcal{Q}(\vec{s}) \leftarrow \bigwedge D_j(\vec{t_j}) \wedge R(t_1,t_2) \wedge B(t) \quad R(x,f(x)) \leftarrow C(x)}{\mathcal{Q}(\vec{s})\sigma \leftarrow \bigwedge D_j(\vec{t_j})\sigma \wedge C(x)\sigma \wedge B(t)\sigma}$$

$\mathcal{T}_1$ σε ένα σύνολο από προτάσεις. Σύμφωνα με τον Πίνακα 2.3 οι προτάσεις που παράγονται λόγω του $\mathcal{T}_1$ είναι οι ακόλουθες:

$$\underline{R(x, f(x))} \leftarrow A(x) \tag{2.1}$$
$$\underline{B(f(x))} \leftarrow A(x) \tag{2.2}$$
$$S(x, y) \leftarrow \underline{R(x, y)} \tag{2.3}$$
$$C(x) \leftarrow \underline{B(x)} \wedge \underline{E(x)} \tag{2.4}$$

Όπως προκύπτει το αξίωμα $A \sqsubseteq \exists R.B$ παράγει δύο προτασεις που περιέχουν το συναρτησιακό σύμβολο f και το οποίο σχετίζεται μοναδικά με την εμφάνιση της έννοιας $\exists R.B$ στο αξίωμα $A \sqsubseteq \exists R.B$.

Στη συνέχεια, ο $\mathcal{I}_{\mathsf{REQ}}$ εφαρμόζεται στο $\mathcal{Q}_1$ και σε όλες τις προτάσεις (2.1)-(2.4), όπου η συνάρτηση επιλογής του Requiem θα επιλέξει τα άτομα που είναι υπογραμμισμένα. Τότε, σύμφωνα με τις πιθανές αλληλεπιδράσεις που φαίνονται στον Πίνακα 2.5, πραγματοποιούνται οι παρακάτω εφαρμογές κανόνων:

$$(2.3), (2.1) \vdash \underline{S(x, f(x))} \leftarrow A(x) \tag{2.5}$$
$$(2.4), (2.2) \vdash C(f(x)) \leftarrow A(x) \wedge \underline{E(f(x))} \tag{2.6}$$
$$\mathcal{Q}_1, (2.5) \vdash Q(x) \leftarrow A(x) \wedge \underline{C(f(x))} \tag{2.7}$$

Αναφερόμενοι στον Πίνακα 2.5 ο πρώτος κανόνας είναι της μορφής 3.1+2.1=2.1, ο δεύτερος της μορφής 3.3+2.3=2.3 και ο τρίτος 1+2.1=1.

Στη συνέχεια, προκύπτει ότι δεν μπορούν να προκύψουν νέες προτάσεις που να διαφέρουν από τις παραχθείσες, λαμβάνοντας υπόψη πιθανές μετονομασίες μεταβλητών από την εφαρμογή του $\mathcal{I}_{\mathsf{REQ}}$. Επομένως, παραλείποντας το βήμα της εκτύλιξης που αναφέραμε, ο αλγόριθμος μπορεί να τερματίσει επιστρέφοντας όλες τις προτάσεις που δε διαθέτουν συναρτήσεις, με άλλα λόγια το σύνολο $\mathcal{R} = \{\mathcal{Q}_1, (2.3), (2.4)\}$. Στην περίπτωση που πραγματοποιηθεί η εκτύλιξη, ο αλγόριθμος εφαρμόζει την πρόταση (2.3) στο $\mathcal{Q}_1$ για να παραχθεί η $\mathcal{Q}_2 = Q(x) \leftarrow R(x, y) \wedge C(y)$, τότε εκτυλίσσει την (2.4) στο $\mathcal{Q}_1$ για να παραχθεί το $\mathcal{Q}_3 = Q(x) \leftarrow S(x, y) \wedge B(x) \wedge E(x)$, και τέλος, την (2.4) στο $\mathcal{Q}_2$ για να παραχθεί το $\mathcal{Q}_4 = Q(x) \leftarrow R(x, y) \wedge B(x) \wedge E(x)$. Σημειώνουμε ότι οι τελευταίες δύο εφαρμογές του κανόνα της εκτύλιξης δεν πραγματοποιούνται στην εκδοχή του Requiem που περιγράφεται στο [110] αλλά όπως αναφέραμε παραπάνω θεωρούμε το βήμα αυτό τετριμμένη επέκταση του αλγορίθμου. Έπειτα, η επαναγραφή θα είναι $\mathcal{R}' = \{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3, \mathcal{Q}_4\}$. ♢

Οι προτάσεις με τύπο 4.1, 4.2 που περιλαμβάνουν το άτομο $C(y)$ λέγονται *RA*-προτάσεις, και προκύπτουν από αξιώματα της μορφής $\exists R.C \sqsubseteq A$, τα *RA-αξιώματα*. Οι υπόλοιπες προτάσεις του Πίνακα 2.4 ονομάζονται *DL-Lite-προτάσεις*, και τα αντίστοιχα αξιώματα DL-Lite-αξιώματα.

Στην DL-Lite οι προτάσεις τύπου 4.1, 4.2 έχουν τη μορφή $B(x) \leftarrow P(x, y)$ και $B(x) \leftarrow P(y, x)$. Συνεπώς, οι κανόνες 4.1+2.1=3.3 έχουν τη μορφή:

$$\frac{B(x) \leftarrow P(x, y) \quad P(x, f(x)) \leftarrow A(x)}{B(x) \leftarrow A(x)}$$

όπου το συμπέρασμα δεν περιλαμβάνει συναρτήσεις. Επιπλέον, σύμφωνα με τον Πίνακα 2.4 η οντολογία εισόδου μπορεί μόνο να περιλαμβάνει μια πρόταση τύπου 3.3 χωρίς συναρτήσεις. Συνεπώς, στην DL-Lite όλες οι προτάσεις με τύπο 3.3 είναι προτάσεις χωρίς συναρτήσεις. Έτσι, ο κανόνας 3.3+2.3=3.3 στον Πίνακα 2.5 δεν εφαρμόζεται ποτέ στην DL-Lite.

KAON2

Στην περίπτωση της Horn-$\mathcal{SHIQ}$ κάποιες προτάσεις, που προκύπτουν από την εφαρμογή των κανόνων της επίλυσης με συναρτησιακά σύμβολα, δεν μπορούν να εξαιρεθούν από το τελικό datalog πρόγραμμα. Το γεγονός αυτό οφείλεται στην εκφραστικότητα της γλώσσας που υποστηρίζει αξιώματα περιορισμού πληθυκότητας. Συγκεκριμένα, λόγω της ύπαρξης αξιωμάτων $A \sqsubseteq \leq 1R.\top$, που μεταφράζονται σε προτάσεις ισότητας, ενδέχεται όροι της μορφής $f(x)$ να αντιστοιχίζονται σε αντικείμενα της βάσης γνώσης. Για παράδειγμα, θεωρούμε μία βάση γνώσης $\mathcal{K} = \{\mathcal{T}, \mathcal{A}\}$, με σώμα ορολογίας $\mathcal{T}$ που περιλαμβάνει τα αξιώματα $A \sqsubseteq \leq 1R, A \sqsubseteq \exists R.C$ και σώμα ισχυρισμών $\mathcal{A} = \{A(a), R(a,b)\}$. Όπως είναι φανερό από το $\mathcal{A}$, το αντικείμενο a ανήκει στην ερμηνεία της A και σχετίζεται μέσω του ρόλου R με το b. Επιπλέον, λόγω του $A \sqsubseteq \exists R.C$ το αντικείμενο b ανήκει στην ερμηνεία της C. Το αξίωμα $A \sqsubseteq \exists R.C$ εισάγει στο μοντέλο της βάσης γνώσης έναν όρο $f(a)$ που σχετίζεται, μέσω του ρόλου R, με το αντικείμενο a. Λόγω του περιορισμού $A \sqsubseteq \leq 1R.\top$, το a δεν μπορεί παρά να συνδέεται με ένα μοναδικό αντικείμενο μέσω του R και έτσι, καταλήγουμε ότι το $f(a)$ ταυτίζεται με το b. Από αυτή την παρατήρηση, συμπεραίνουμε ότι το b ανήκει στην ερμηνεία της C. Το συμπέρασμα προκύπτει κατά τη διαδικασία επαναγραφής, όταν παράγεται μία πρότασης της μορφής $f(a) \approx b$ από τις $R(a,b)$ και $f(x) \approx z \leftarrow R(x,z)$. Όπως γίνεται κατανοητό, η πρόταση $f(x) \approx z \leftarrow R(x,z)$ ενώ διαθέτει συναρτησιακό σύμβολο, πρέπει να ληφθεί υπόψη κατά την κατασκευή του datalog προγράμματος.

Στη συνέχεια, περιγράφουμε τον αλγόριθμο επαναγραφής για οντολογίες διατυπωμένες σε Horn-$\mathcal{SHIQ}$ έτσι όπως έχει υλοποιηθεί στο σύστημα KAON2 [70]. Σημειώνουμε ότι ο αλγόριθμος του συστήματος KAON2 μπορεί να υπολογίσει όλες τις βασικές προτάσεις που εξάγονται από μία οντολογία $\mathcal{T}$ και ένα ABox $\mathcal{A}$. Κατασκευάζει την επαναγραφή ενός Horn-$\mathcal{SHIQ}$ TBox και ενός ερωτήματος που περιλαμβάνει ένα μόνο άτομο, και δεν είναι δηλαδή συζευκτικό ερώτημα. Περιγράφεται από τα παρακάτω βήματα:

1. *Μετατροπή σε προτάσεις*: Η οντολογία εισόδου μεταφράζεται όπως και στην περίπτωση του Requiem, σε ένα σύνολο προτάσεων λογικής πρώτης τάξης αντικαθιστώντας τις υπαρξιακές μεταβλητές με συναρτήσεις skolem [9].

2. *Κορεσμός*: Στο σύνολο των προτάσεων που κατασκευάστηκαν στο προηγούμενο βήμα εφαρμόζονται εξαντλητικά οι κανόνες της βασικής υπέρθεσης που παραμετροποιείται από μία διάταξη $\succ$, και μία συνάρτηση επιλογής, που επιλέγει τα αρνητικά άτομα με πληθυκότητα δύο, δηλαδή τους ρόλους που βρίσκονται στο σώμα μίας πρότασης.

3. *Τελική επεξεργασία*: Το κορεσμένο σύνολο προτάσεων μετασχηματίζεται σε ένα ισοδύναμο πρόγραμμα χωρίς συναρτήσεις, από το οποίο κατασκευάζεται το datalog πρόγραμμα της εξόδου.

Παράδειγμα 2.4.4. Έστω το παρακάτω Horn-$\mathcal{SHIQ}$ TBox $\mathcal{T}$ αποτελείται από τα αξιώματα αριστερα τα οποία μεταφράζονται στις προτάσεις δεξιά:

$$\top \sqsubseteq \leq 1R.B \quad \rightsquigarrow \quad y \approx z \leftarrow R(x,y) \wedge R(x,z) \wedge B(y) \wedge B(z) \tag{2.8}$$

$$K \sqsubseteq \exists R.B \quad \rightsquigarrow \quad R(x, f(x)) \leftarrow K(x) \tag{2.9}$$

$$B(f(x)) \leftarrow K(x) \tag{2.10}$$

$$M \sqsubseteq \exists R^{-} \quad \rightsquigarrow \quad R(h(x), x) \leftarrow M(x) \tag{2.11}$$

Ο λογισμός που έχει υλοποιηθεί στο [70] εφαρμόζει τον παρακάτω κανόνα υπερεπίλυσης (hyperresolution):

$$(2.8), (2.9), (2.11) \vdash f(h(x)) \approx x \leftarrow K(h(x)) \wedge M(x) \wedge B(f(h(x))) \wedge B(x) \tag{2.12}$$

Στη συνέχεια, το άτομο $B(f(h(x)))$ είναι μέγιστο στην πρόταση (2.12) με βάση την διάταξη $\succ$, ενώ δεν υπάρχει κάποιο επιλεγμένο άτομο στο σώμα της πρότασης. Σαν αποτέλεσμα, εφαρμόζεται ο παρακάτω κανόνας επίλυσης:

$$(2.12), (2.10) \vdash f(h(x)) \approx x \leftarrow K(h(x)) \wedge M(x) \wedge B(x) \tag{2.13}$$

Εφόσον η πρόταση (2.13) δεν περιλαμβάνει επιλεγμένα άτομα και το άτομο στην κεφαλή είναι μέγιστο, εφαρμόζεται ο κανόνας της βασικής υπέρθεσης με την πρόταση (2.9):

$$(2.9), (2.13) \vdash R(h(x), x) \leftarrow K(h(x)) \wedge M(x) \wedge B(x) \tag{2.14}$$

Τέλος, η διαδικασία κορεσμού τερματίζει καθώς δεν εφαρμόζονται άλλοι κανόνες. Ο αλγόριθμος απορρίπτει τις προτάσεις (2.13) και (2.14) προκειμένου να κατασκευάσει το datalog πρόγαμμα εξόδου. $\diamondsuit$

Στις επόμενες ενότητες θεωρούμε για λόγους απλότητας, ότι το σώμα ισχυρισμών, TBox, δεν περιλαμβάνει αξιώματα ΠΛ αλλά προτάσεις λογικής πρώτης τάξης.

Μέρος II

Επαναγραφή Συζευκτικών Ερωτημάτων

Κεφάλαιο 3

Επαναγραφή με επίλυση στην DL-Lite

Στην παρούσα ενότητα παρουσιάζουμε τον αλγόριθμό μας, που βασίζεται στην τεχνική της επίλυσης, για την DL-Lite. Αρχικά, επεξηγούμε τα βασικά μειονεκτήματα των αλγορίθμων επαναγραφής που βασίζονται στην επίλυση και τα οποία αποτέλεσαν κίνητρο για το σχεδιασμό του νέου αλγορίθμου. Έπειτα, παρουσιάζουμε το λογισμό μας και τελικά, αποδεικνύουμε την ορθότητά του.

3.1 Κανόνες επίλυσης του $\mathcal{I}_{lite}$

Παράδειγμα 3.1.1. Θεωρούμε το σώμα ορολογίας $\mathcal{T}_1$ και το ερώτημα $\mathcal{Q}_1$ από το παράδειγμα 2.4.3 καθώς και τους κανόνες συμπερασμού που πραγματοποιούνται από το $\mathcal{I}_{\mathsf{REQ}}$ στο $\mathcal{T}_1 \cup \{\mathcal{Q}_1\}$.

Όπως μπορούμε να παρατηρήσουμε ο αλγόριθμος εφάρμοσε τρεις περιττούς κανόνες. Πιο συγκεκριμένα, όλοι οι κανόνες που παράγουν τις προτάσεις (2.5)–(2.7) είναι περιττοί. Πράγματι, οι συγκεκριμένες προτάσεις απορρίπτονται στο τελικό βήμα εφόσον περιέχουν συναρτήσεις και άρα δεν είναι δυνατό να συμπεριληφθούν στο datalog πρόγραμμα, ενώ την ίδια στιγμή δεν χρησιμοποιούνται από τους επόμενους κανόνες για την παραγωγή κάποιας πρότασης που ανήκει στην επαναγραφή $\mathcal{R}$.

Θεωρούμε τώρα ότι το $\mathcal{T}_1$ περιλαμβάνει επιπρόσθετα την $C(x) \leftarrow B(x)$. Η πρόταση αυτή αναλύεται με την (2.2) και παράγει την $C(f(x)) \leftarrow A(x)$ η οποία εν συνεχεία, αναλύεται με την (2.7) για να προκύψει το ερώτημα $Q(x) \leftarrow A(x)$. Εφόσον το ερώτημα δεν περιλαμβάνει συναρτήσεις, αποτελεί μέλος του τελικού συνόλου επαναγραφής για το $\mathcal{Q}_1$ με βάση το $\mathcal{T}_1 \cup \{C(x) \leftarrow B(x)\}$. Προφανώς, στη συγκεκριμένη περίπτωση οι κανόνες που παράγουν τις προτάσεις (2.2) και (2.7) δεν είναι περιττοί. $\diamondsuit$

Όπως έχει μελετηθεί [122], οι οντολογίες που έχουν αναπτυχθεί για εφαρμογές του πραγματικού κόσμου, τυπικά περιλαμβάνουν πολλές προτάσεις της μορφής $R(x, f_i(x)) \leftarrow A_i(x)$. Οι συγκεκριμένες μαζί άλλες, όπως η (2.3) και το $\mathcal{Q}_1$ του παραδείγματος 2.4.3, υπονοούν ότι οι τυπικοί αλγόριθμοι που βασίζονται στην επίλυση παράγουν πολλές προτάσεις της μορφής $S(x, f_i(x)) \leftarrow A_i(x)$ και $Q(x) \leftarrow A(x) \wedge C(f_i(x))$.

Όπως είναι προφανές, σε μεγάλες οντολογίες οι προτάσεις αυτής της μορφής είναι περιττές, όπως εξηγήσαμε στο προηγούμενο παράδειγμα, και η παραγωγή τους μπορεί να δυσχεραίνει σημαντικά την επίδοση των αλγορίθμων επαναγραφής.

Στο παράδειγμα που ακολουθεί φαίνεται πως η παραγωγή προτάσεων με συναρτησιακά σύμβολα μπορεί να οδηγήσει στην παραγωγή της ίδιας πρότασης περισσότερες φορές.

Παράδειγμα 3.1.2. Θεωρούμε το ερώτημα $\mathcal{Q}_3 = Q(x) \leftarrow S(x,y) \wedge P(y,z) \wedge D(z)$ και το σώμα ορολογίας $\mathcal{T}_2 = \mathcal{T}_1 \cup \{\mathcal{C}_1, \mathcal{C}_2\}$, όπου το $\mathcal{T}_1$ ορίζεται στο παράδειγμα 2.4.3 και οι προτάσεις $\mathcal{C}_1, \mathcal{C}_2$ έχουν ως εξής:

$$\begin{aligned} \mathcal{C}_1 &= P(x, g(x)) \leftarrow B(x) \\ \mathcal{C}_2 &= D(g(x)) \leftarrow B(x) \end{aligned}$$

Το σύστημα $\mathcal{I}_{\mathsf{REQ}}$ εφαρμόζει τους παρακάτω κανόνες:

$$\begin{aligned} \mathcal{Q}_3, (2.5) &\vdash Q(x) \leftarrow A(x) \wedge P(f(x), z) \wedge D(z) && (3.1) \\ (3.1), \mathcal{C}_1 &\vdash Q(x) \leftarrow A(x) \wedge B(f(x)) \wedge D(g(f(x))) && (3.2) \\ (3.2), \mathcal{C}_2 &\vdash Q(x) \leftarrow A(x) \wedge B(f(x)) && (3.3) \\ (3.3), (2.2) &\vdash Q(x) \leftarrow A(x) && (3.4) \\ \mathcal{Q}_3, \mathcal{C}_1 &\vdash Q(x) \leftarrow S(x,y) \wedge B(y) \wedge D(g(y)) && (3.5) \\ (3.5), \mathcal{C}_2 &\vdash Q(x) \leftarrow S(x,y) \wedge B(y) && (3.6) \\ (3.6), (2.5) &\vdash (3.3) \end{aligned}$$

Η κατασκευή των προτάσεων (3.1) εως (3.3) και η κατασκευή όλων των προτάσεων μετά την (3.5) αντιστοιχίζονται σε δύο διαφορετικά μονοπάτια παραγωγής της ίδιας πρότασης, $Q(x) \leftarrow A(x)$. Είναι προφανές ότι το δεύτερο μονοπάτι είναι προτιμότερο καθώς επιπρόσθετα οδηγεί στην παραγωγή της πρότασης (3.6) που δεν περιλαμβάνει συναρτήσεις. ♢

Μία αρχική προσέγγιση επίλυσης των παραπάνω ζητήματων θα μπορούσε να εφαρμόζει το λογισμό $\mathcal{I}_{\mathsf{REQ}}$ ως εξής: αρχικά, γίνεται εξαντλητική εφαρμογή των κανόνων στο $\mathcal{T}$ για να προκύψει ένα νέο σύνολο προτάσεων $\mathcal{T}_{sat}$. Έπειτα ένας κανόνας που πρόκειται να παράγει μία πρόταση που περιλαμβάνει άτομο με συνάρτηση, εφαρμόζεται μόνο στην περίπτωση που το $\mathcal{T}_{sat}$ περιλαμβάνει κάποια πρόταση ικανή να απομακρύνει αυτό το άτομο σε επόμενο βήμα. Στο Παράδειγμα 2.4.3, αυτή τη βελτιστοποιημένη στρατηγική θα ανέλυε το $\mathcal{Q}_1$ με (2.5) μόνο αν η $C(f(x)) \leftarrow A(x)$ υπάρχει στο $\mathcal{T}_{sat}$. Εάν συμβαίνει κάτι τέτοιο, είναι εξαρχής γνωστό ότι αφού παραχθεί η (2.7), που περιλαμβάνει το άτομο $C(f(x))$ στο σώμα, η πρόταση $C(f(x)) \leftarrow A(x)$ μπορεί να χρησιμοποιηθεί σα δευτερεύουσα προκείμενη στην εφαρμογή του κανόνα που θα απομακρύνει το $C(f(x))$ από την (2.7). Σαν αποτέλεσμα, παράγεται η $Q(x) \leftarrow A(x)$ που δεν περιέχει συναρτήσεις. Στην πραγματικότητα οι δύο αυτοί κανόνες μπορούν να συγχωνευτούν σε ένα κανόνα επίλυσης με κύρια πρόταση την $\mathcal{Q}_1$ και δευτερεύουσες τις (2.5) και $C(f(x)) \leftarrow A(x)$, δηλαδή δύο προτάσεις με την ίδια συνάρτηση στην

κεφαλή. Στο παράδειγμα 3.1.2, η στρατηγική αυτή θα μπορούσε να παράγει μόνο τις προτάσεις (3.6) και (3.4)του δεύτερου μονοπατιού ενώ, θα αγνοούσε εντελώς το πρώτο μέρος.

Παρόλα αυτά, προκειμένου να υλοποιηθεί η παραπάνω προσέγγιση, χρειάζεται να διατρέξουμε τπ σύνολο $\mathcal{T}_{sat}$ εκθετικά πολλές φορές, προκειμένου να αναζητήσουμε ζεύγη προτάσεων που συνδυαστικά εισάγουν στο ένα βήμα και απομακρύνουν στο επόμενο, κάποιον όρο που περιλαμβάνει συναρτησιακό σύμβολο. Όμως το $\mathcal{T}_{sat}$ μπορεί να είναι τετραγωνικά μεγαλύτερο από το $\mathcal{T}$ και άρα η προσέγγιση αυτή δεν αποδίδει στην πράξη.

Η δυσκολία στην αποδοτική υλοποίηση του παραπάνω βήματος επίλυσης οφείλεται κυρίως στο γεγονός ότι ο λογισμός του $\mathcal{I}_{\mathsf{REQ}}$ ακολουθεί μία τεχνική "από πάνω προς τα κάτω" κατά την εφαρμογή του κανόνα της επίλυσης που παράγει νέες προτάσεις με συναρτήσεις. Για παράδειγμα, η πρόταση (2.5) αναπτύχθηκε διαδίδοντας την f από την πρόταση (2.1) στην (2.3) προκαλώντας αύξηση στο μέγεθος του $\mathcal{T}_{sat}$. Προκειμένου να εφαρμόσουμε αποδοτικά το προαναφερθέν βήμα επίλυσης χρειάζεται να επιλέξουμε ζεύγη προτάσεων από το $\mathcal{T}$ και όχι από το $\mathcal{T}_{sat}$. Για να πετύχουμε κάτι τέτοιο προτείνουμε μία "από κάτω προς τα πάνω" τεχνική κατευθυνόμενη από το στόχο που μοιάζει με τις SLD παραγωγές.

Παράδειγμα 3.1.3. Έστω το $\mathcal{Q}_1$ και το $\mathcal{T}_1$ από το Παράδειγμα 3.1.1. Έστω επίσης το σώμα ορολογίας $\mathcal{T}' = \mathcal{T}_1 \cup \{C(x) \leftarrow B(x)\}$.

Θεωρούμε ότι έχουμε ένα λογισμό ο οποίος αντί να εφαρμόσει τον κανόνα της επίλυσης στις προτάσεις (2.1) και (2.3), όπως συνέβει με το $\mathcal{I}_{\mathsf{REQ}}$ στο Παράδειγμα 2.4.3, γεγονός που μεταφέρει τη συνάρτηση f στην παραγόμενη πρόταση, αναλύει το $\mathcal{Q}_1$ με (2.3), και παράγει την $\mathcal{Q}_2 = Q(x) \leftarrow R(x,y) \wedge C(y)$. Η $\mathcal{Q}_2$ δεν αναλύεται με την πρόταση (2.1) γιατί τότε θα προέκυπτε η $Q(x) \leftarrow A(x) \wedge C(f(x))$, ενώ στο $\mathcal{T}'$ δεν υπάρχει κάποια πρόταση της μορφής $C(f(x)) \leftarrow A(x)$ που να απομακρύνει το συναρτησιακό σύμβολο $f(x)$. Έπειτα, η πρόταση $\mathcal{Q}_2$ αναλύεται με την $C(x) \leftarrow B(x)$ παράγοντας τη $\mathcal{Q}_3 = Q(x) \leftarrow R(x,y) \wedge B(y)$. Στη συνέχεια, η $\mathcal{Q}_3$ μπορεί να αναλυθεί με την (2.1) εφόσον υπάρχει η πρόταση (2.2) στο $\mathcal{T}'$, οπότε τελικά παράγεται η $Q(x) \leftarrow A(x)$ που δεν περιέχει συναρτήσεις. $\diamondsuit$

Η κρίσιμη διαφορά ανάμεσα στο λογισμό του $\mathcal{I}_{\mathsf{REQ}}$ και τον υποθετικό μας λογισμό είναι ότι ο τελευταίος προκειμένου να εφαρμόσει τον κανόνα της επίλυσης, επιλέγει προτάσεις από το σώμα ορολογίας της εισόδου και όχι από το κορεσμένο σώμα ορολογίας. Πέραν από το πολύ μικρότερο πλήθος προτάσεων σε σύγκριση με το κορεσμένο σύνολο, το σώμα ορολογίας της εισόδου περιλαμβάνει το πολύ δύο προτάσεις που διαθέτουν την ίδια συνάρτηση. Αυτό γίνεται κατανοητό αν παρατηρήσουμε ότι τα συναρτησιακά σύμβολα είναι μοναδικά για κάθε αξίωμα μορφής $\exists R.B$, ενώ ο λογισμός δεν παράγει νέες προτάσεις σε συναρτήσεις. Στην πράξη, η εύρεση των ζευγών από προτάσεις που θα χρησιμοποιηθούν στο βήμα της επίλυσης μπορεί να πραγματοποιηθεί με τη χρήση κατάλληλων δεικτών.

Όλες οι προηγούμενες παρατηρήσεις αποτέλεσαν το κίνητρο για το σχεδιασμό του αλγόριθμου επαναγραφής μας που βασίζεται στην επίλυση, για τη γλώσσα DL-Lite, τον οποίο παρουσιάζουμε στη συνέχεια.

Ορισμός 3.1.4. Έστω $\mathcal{Q}$ ένα ΣΕ, και μία πρόταση $\mathcal{C}_{(i)}$ σε DL-Lite. Συμβολίζουμε με $\mathcal{I}_{lite}$ το σύστημα συμπερασμού που αποτελείται από τους κανόνες του Πίνακα 3.1. Για ένα ΣΕ $\mathcal{Q}$ και ένα σώμα ορολογίας $\mathcal{T}$ σε DL-Lite ορίζουμε ως Rapid-Lite$(\mathcal{Q}, \mathcal{T})$ όλες τις προτάσεις χωρίς συνάρτηση που παράγονται απο το $\mathcal{T} \cup \{\mathcal{Q}\}$ με το λογισμό $\mathcal{I}_{lite}$. △

Πίνακας 3.1: Οι κανόνες του λογισμού $\mathcal{I}_{lite}$

εκτύλιξη (*unfolding*): $\dfrac{\mathcal{Q} \quad \mathcal{C}}{\mathcal{Q}'\sigma}$

όπου αν $x \mapsto f(y) \in \sigma$, τότε $x \in$ unbound$(\mathcal{Q})$.

συρρίκνωση (*shrinking*): $\dfrac{\mathcal{Q} \quad \mathcal{C}_1\ [\mathcal{C}_2]}{\mathcal{Q}'\sigma}$

όπου $\mathcal{Q}'\sigma$ είναι συμπέρασμα χωρίς συναρτήσεις των $\mathcal{Q}, \mathcal{C}_1$, και $\mathcal{C}_2$, και υπάρχει κάποιο $x \mapsto f(y) \in \sigma$ τέτοιο ώστε $x \notin$ unbound$(\mathcal{Q})$.

Ο κανόνας εκτύλιξης αντιστοιχίζεται σε κανόνες της (κλασικής) δυαδικής επίλυσης όπου η συμπερασματική πρόταση δεν περιλαμβάνει συναρτήσεις. Αυτό επιτυγχάνεται από τη συνθήκη στην αντικατάσταση σ. Αντίθετα, η συρρίκνωση είναι ένας κανόνας που συγκεντρώνει πολλούς κανόνες σε έναν, αντιστοιχίζεται σε παραγωγή της μορφής $\mathcal{Q}, \mathcal{Q}_1, \ldots, Q_n, \mathcal{Q}'$, όπου η $\mathcal{Q}$, δε διαθέτει συναρτήσεις, οι $\mathcal{Q}_1, \ldots, \mathcal{Q}_n$ περιλαμβάνουν ένα συναρτησιακό σύμβολο f το οποίο οι διαδοχικοί κανόνες της επίλυσης επιχειρούν να απομακρύνουν προκειμένου να παραχθεί τελικά η $\mathcal{Q}'$ που δεν περιλαμβάνει συναρτήσεις. Σημειώνουμε ότι, εφόσον το $\mathcal{Q}$ δεν περιλαμβάνει συναρτήσεις και το $\mathcal{Q}_1$ αναφέρει την f, ο μέγιστος κοινός ενοποιητής αντιστοιχίζει μια δευσμευμένη μεταβλητή του $\mathcal{Q}$ σε όρο της μορφής $f(y)$. Επιπλέον, όπως προαναφέρθηκε στο $\mathcal{T}$ υπάρχουν το πολύ δύο προτάσεις που αναφέρουν το f. Έτσι, ο κανόνας της συρρίκνωσης μπορεί να εφαρμόζει το πολύ δύο δευτερεύουσες προκείμενες, της μορφής $R(x, f(x)) \leftarrow B(x)$ και $A(f(x)) \leftarrow B(x)$. Τέλος, σημειώνουμε ότι αφού ο λογισμός παράγει μόνο προτάσεις της μορφής 1 οι δευτερεύουσες προτάσεις ανήκουν πάντα στο $\mathcal{T}$.

Εάν ο λογισμός $\mathcal{I}_{lite}$ εφαρμοστεί στα $\mathcal{T}'$ και $\mathcal{Q}_1$ του Παραδείγματος 3.1.3, θα πραγματοποιήσει τα βήματα που περιγράψαμε στο παράδειγμα. Δηλαδή, το $\mathcal{Q}_2$ παράγεται εφαρμόζοντας τον κανόνα της εκτύλιξης στα $\mathcal{Q}_1$ και (2.3), το $\mathcal{Q}_3$ παράγεται ξανά με εκτύλιξη στα $\mathcal{Q}_2$ και $C(x) \leftarrow B(x)$, ενώ το $Q(x) \leftarrow A(x)$ προκύπτει με εφαρμογή του κανόνα συρρίκνωσης στα $\mathcal{Q}_3$, (2.2), και (2.1).

Παρουσιάζουμε στη συνέχεια, την αλγοριθμική διαδικασία για τον υπολογισμό της επαναγραφής στην DL-Lite. Όπως φαίνεται, δεδομένου ενός συζευκτικού ερωτήματος, $\mathcal{Q}$, και ενός σώματος ορολογίας, $\mathcal{T}$, εφαρμόζονται οι κανόνες εκτύλιξης και συρρίκνωσης με τη διαδικασία RLite$(\mathcal{Q}, \mathcal{T})$ (γραμμή 3). Κάθε ερώτημα που παράγεται προστίθεται στο σύνολο $\mathcal{R}_Q$. Στη συνέχεια, καλείται εκ νέου η RLite$(\mathcal{Q}', \mathcal{T})$ και εφαρμόζονται οι κανόνες της επίλυσης, για ένα νέο ερώτημα $\mathcal{Q}'$ από το $\mathcal{R}_Q$ (γραμμή 2). Η ίδια διαδικασία επαναλαμβάνεται, μέχρις ότου να μην πράγεται κάποια νέα πρό-

ταση. Το σύνολο $\mathcal{R}_Q$ αποτελεί την επαναγραφή Rapid-Lite$(\mathcal{Q}, \mathcal{T})$ που επιστρέφει ο αλγόριθμος στην έξοδό του.

Αλγόριθμος 1 Αλγόριθμος παραγωγής της επαναγραφής Rapid-Lite$(\mathcal{Q}, \mathcal{T})$

Είσοδος: ένα ΣΕ, $\mathcal{Q}$ και ένα TBox, $\mathcal{T}$ που περιλαμβάνει προτάσεις.

1: **repeat**
2: Επίλεξε μία πρόταση $\mathcal{Q}' \in \mathcal{R}_Q$
3: $\mathcal{R}_Q := \mathcal{R}_Q \cup$ RLite$(\mathcal{Q}', \mathcal{T})$
4: **until** δεν προστίθεται κανένα νέο ερώτημα στο $\mathcal{R}_Q$ λαμβάνοντας υπόψη και τις μετονομασίες των μεταβλητών
5: **return** $\mathcal{R}_Q$

3.2 Ορθότητα του $\mathcal{I}_{lite}$

Προκειμένου να δείξουμε την ορθότητα του αλγορίθμου επαναγραφής $\mathcal{I}_{lite}$, θα δείξουμε ότι μία παραγωγή που προκύπτει από το λογισμό $\mathcal{I}_{\text{REQ}}$, είναι δυνατό να μετατραπεί σε μία παραγωγή από το $\mathcal{I}_{lite}$. Έτσι, εφόσον, ο $\mathcal{I}_{\text{REQ}}$ λογισμός είναι πλήρης, θα έχουμε αποδείξει ότι εφαρμόζοντας εξαντλητικά τους κανόνες του $\mathcal{I}_{lite}$ στο $\mathcal{T} \cup \{\mathcal{Q}\}$ παράγουμε όλα τα απαραίτητα μέλη της επαναγραφής. Οργανώνουμε την απόδειξή μας σε δύο σκέλη. Πρώτον, δείχνουμε ότι κάθε παραγωγή από το Requiem μπορεί να μετατραπεί σε μία SLD παραγωγή. Για το σκοπό αυτό δείχνουμε πως ένας κανόνας του Requiem τη μορφής $\mathcal{Q}, \mathcal{C} \vdash \mathcal{Q}'$ όπου $\mathcal{T} \vdash_i \mathcal{C}$, μπορεί να "ξεδιπλωθεί" σε δύο κανόνες της μορφής $\mathcal{Q}, \mathcal{C}_1 \vdash \mathcal{Q}''$, $\mathcal{Q}'', \mathcal{C}_2 \vdash \mathcal{Q}'$ όπου $\mathcal{T} \vdash_{i-1} \mathcal{C}_1, \mathcal{T} \vdash_{i-1} \mathcal{C}_2$ και $\mathcal{C}_1, \mathcal{C}_2 \vdash \mathcal{C}$. Έτσι, από επαναλαμβανόμενες εφαρμογές του ισχυρισμού προκύπτουν κανόνες με δευτερεύουσες προτάσεις από το $\mathcal{T}$, δηλαδή μία SLD παραγωγή. Δεύτερον, δείχνουμε πως μία SLD παραγωγή μπορεί να μετατραπεί σε μία παραγωγή από το $\mathcal{I}_{lite}$.

Λήμμα 3.2.1. *Έστω $\mathcal{T}$ ένα DL-Lite σώμα ορολογίας και έστω $\mathcal{Q}$ ένα ΣΕ. Κάθε πρόταση τύπου 1 που προκύπτει από τα $\mathcal{T} \cup \{\mathcal{Q}\}$ από το λογισμό $\mathcal{I}_{\text{REQ}}$ μπορεί επίσης να προκύψει από το $\mathcal{T} \cup \{\mathcal{Q}\}$ από το $\mathcal{I}_{\text{SLD}}$.*

Απόδειξη. Αποδεικνύουμε επαγωγικά ότι για κάθε πρόταση $\mathcal{Q}'$ τύπου 1 τέτοια ώστε $\mathcal{T}, \mathcal{Q} \vdash_i^{\mathcal{I}_{\text{REQ}}} \mathcal{Q}'$ (που παράγεται δηλαδή σε βάθος i) ισχύει επίσης ότι $\mathcal{T}, \mathcal{Q} \vdash^{\mathcal{I}_{\text{SLD}}} \mathcal{Q}'$.

Αρχική περίπτωση για i=0: Σε αυτή την περίπτωση το $\mathcal{Q}'$ είναι το ΣΕ της εισόδου $\mathcal{Q}$ και έτσι έχουμε ξεκάθαρα ότι $\mathcal{T}, \mathcal{Q} \vdash^{\mathcal{I}_{\text{SLD}}} \mathcal{Q}$.

Επαγωγικό βήμα: Υποθέτουμε ότι για κάθε $\mathcal{Q}_i$ τέτοιο ώστε $\mathcal{T}, \mathcal{Q} \vdash_i^{\mathcal{I}_{\text{REQ}}} \mathcal{Q}_i$ έχουμε $\mathcal{T}, \mathcal{Q} \vdash^{\mathcal{I}_{\text{SLD}}} \mathcal{Q}_i$ (επαγωγική υπόθεση). Υποθέτουμε τώρα ότι σε επόμενο βήμα παράγεται μία πρόταση $\mathcal{Q}_{i+1}$ τύπου 1, δηλαδή $\mathcal{T}, \mathcal{Q} \vdash_{i+1}^{\mathcal{I}_{\text{REQ}}} \mathcal{Q}_{i+1}$. Από τον ορισμό του $\mathcal{I}_{\text{REQ}}$ η $\mathcal{Q}_{i+1}$ παράγεται από ένα κανόνα της μορφής $\mathcal{Q}_i, \mathcal{C} \vdash^{\mathcal{I}_{\text{REQ}}} \mathcal{Q}_{i+1}$, που έχει δηλαδή ως κύρια πρόταση μία πρόταση τύπου 1 τέτοια ώστε $\mathcal{T}, \mathcal{Q} \vdash_i^{\mathcal{I}_{\text{REQ}}} \mathcal{Q}_i$ και ως δευτερεύουσα πρόταση μια $\mathcal{C}$ τέτοια ώστε $\mathcal{T} \vdash^{\mathcal{I}_{\text{REQ}}} \mathcal{C}$. Προφανώς, $\mathcal{Q}_i, \mathcal{C} \vdash^{\mathcal{I}_{\text{SLD}}} \mathcal{Q}_{i+1}$. Εάν $\mathcal{C} \in \mathcal{T}$ τότε, από τα προηγούμενα και την επαγωγική υπόθεση προκύπτει αμέσως ότι

$\mathcal{T}, \mathcal{Q} \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}_{i+1}$. Αλλιώς, έχουμε $\mathcal{T} \vdash_{j}^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{C}$ με $j > 0$, και αρκεί να δείξουμε ότι η $\mathcal{Q}_{i+1}$ μπορεί επίσης να παραχθεί από το $\mathcal{Q}_i$ με $\mathcal{I}_{\mathsf{SLD}}$ χρησιμοποιώντας μόνο προτάσεις του $\mathcal{T}$. Αποδεικνύουμε για το σκοπό αυτό τον παρακάτω ισχυρισμό:

Ισχυρισμός 1: Για κάθε $\mathcal{Q}_{i+1}$ και $\mathcal{C}$ τέτοια ώστε $\mathcal{Q}_i, \mathcal{C} \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}_{i+1}$, $\mathcal{T} \vdash_{j}^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{C}$, και $j > 0$ υπάρχει μία $\mathcal{C}_1$ και $\mathcal{C}_2$ ώστε να ισχύουν τα παρακάτω:

- $\mathcal{T} \vdash_{j-1}^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{C}_1$ και $\mathcal{T} \vdash_{j-1}^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{C}_2$, και
- $\mathcal{Q}_i, \mathcal{C}_1 \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}'$, $\mathcal{Q}', \mathcal{C}_2 \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}_{i+1}$.

Δείχνουμε τον Ισχυρισμό 1 κάνοντας ανάλυση των περιπτώσεων για τους διαφορετικούς τύπους προτάσεων που μπορούν να προκύψουν από το $\mathcal{T}$ με βάση το $\mathcal{I}_{\mathsf{REQ}}$. Σύμφωνα με το $\mathcal{I}_{\mathsf{REQ}}$, μόνο προτάσεις με τύπο 2.1, 2.2, 2.3 (της μορφής $B(f(x)) \leftarrow C(x) \wedge [\mathbf{A}(f(x))]$), ή 3.3 (της μορφής $A(x) \leftarrow B(x)$) μπορούν να εξαχθούν από το $\mathcal{T}$ με βάση το $\mathcal{I}_{\mathsf{REQ}}$. Εξετάζουμε την κάθε περίπτωση ξεχωριστά:

1. Η $\mathcal{C}$ είναι τύπου 2.1. Τότε ο $\mathcal{Q}_i, \mathcal{C} \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}_{i+1}$ είναι της μορφής:

$$\frac{Q_P(\vec{s}) \leftarrow \bigwedge D_j(\vec{t_j}) \wedge R(v,u) \quad R(x, f(x)) \leftarrow A(x)}{Q_P(\vec{s})\sigma \leftarrow \bigwedge D_j(\vec{t_j})\sigma \wedge A(x)\sigma}$$

   όπου σ είναι ένας μκε για τα άτομα $\{R(v,u), R(x, f(x))\}$. Επιπλέον, ο $\mathcal{I}_{\mathsf{REQ}}$ παράγει προτάσεις τύπου 2.1 αναλύοντας τις προτάσεις $\mathcal{C}_1, \mathcal{C}_2$ με τύπο 3.1, 2.1, ή 3.2, 2.2, αντίστοιχα. Δείχνουμε την περίπτωση που η $\mathcal{C}_1$ είναι τύπου 3.1 ($R(x,y) \leftarrow P(x,y)$) και η $\mathcal{C}_2$ τύπου 2.1 ($P(x, f(x)) \leftarrow A(x)$), η περίπτωση των τύπων 3.2 και 2.2 είναι παρόμοια. Υποθέτουμε ότι ο μκε κατά την εφαρμογή του κανόνα συμπερασμού στις $\mathcal{C}_1$, $\mathcal{C}_2$ είναι $\sigma_1 = \{y/f(x)\}$. Προφανώς, το $\mathcal{Q}_i$ αναλύεται με την $\mathcal{C}_1$ παράγοντας την πρόταση:

$$\mathcal{Q}' = Q_P(\vec{s})\theta \leftarrow \bigwedge D_j(\vec{t_j})\theta \wedge P(x,y)\theta$$

   όπου θ είναι ο μκε για τα $\{R(v,u), R(x,y)\}$. Η αντιστοίχιση $\theta_1 = \{x \mapsto v, y \mapsto u\}$ είναι ένας μκε για τα $\{R(v,u), R(x,y)\}$. Παρόλα αυτά αν οι v, u είναι μεταβλητές τότε $\theta_2 = \{v \mapsto x, u \mapsto y\}$ είναι επίσης ένας πιθανός μκε. Όμως οι προτάσεις που προκύπτουν από την εφαρμογή των θ_1 και θ_2 είναι ισοδύναμες λαμβάνοντας υπόψη τη μετονομασία των μεταβλητών. Έτσι, χωρίς βλάβη της γενικότητας μπορούμε να υποθέσουμε ότι $\theta = \theta_1$. Επιπλέον, αφού οι x, y δεν εμφανίζονται στο $\mathcal{Q}_i$, η πρόταση $\mathcal{Q}'$ είναι στην πραγματικότητα της μορφής $\mathcal{Q}' = Q_P(\vec{s}) \leftarrow \bigwedge D_j(\vec{t_j}) \wedge P(v,u)$. Συμπερασματικά, αφού οι μεταβλητές που εμφανίζονται στο $P(v,u)$ είναι οι ίδιες με εκείνες στο $R(v,u)$, η $\mathcal{Q}'$ αναλύεται με $\mathcal{C}_2$ με μκε σ για να προκύψει η πρόταση $\mathcal{Q}_{i+1}$. Επομένως, υπάρχουν κανόνες της μορφής $\mathcal{Q}_i, \mathcal{C}_1 \vdash \mathcal{Q}'$ και $\mathcal{Q}', \mathcal{C}_2 \vdash \mathcal{Q}_{i+1}$. Εφόσον ο $\mathcal{I}_{\mathsf{REQ}}$ δεν παράγει ποτέ προτάσεις τύπου 3.1, τότε $\mathcal{C}_1 \in \mathcal{T}$ και για την $\mathcal{C}_2$ έχουμε ότι $\mathcal{T} \vdash_{j-1}^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{C}_2$.

2. Η $\mathcal{C}$ είναι τύπου 2.2. Η περίπτωση αυτή είναι συμμετρική της προηγούμενης.

3. Η $\mathcal{C}$ είναι τύπου 2.3. Τότε, ο κανόνας $\mathcal{Q}_i, \mathcal{C} \vdash^{\mathcal{I}_{\text{SLD}}} \mathcal{Q}_{i+1}$ έχει την ακόλουθη μορφή:

$$\frac{Q_P(\vec{s}) \leftarrow \bigwedge D_j(\vec{t_j}) \wedge B(v) \quad B(f(x)) \leftarrow A(x) \wedge [\mathbf{A}'(f(x))]}{Q_P(\vec{s})\sigma \leftarrow \bigwedge D_j(\vec{t_j})\sigma \wedge A(x)\sigma \wedge [\mathbf{A}'(f(x))\sigma]}$$

όπου σ είναι ένας μκε για τα $\{B(v), B(f(x))\}$. Επιπλέον, ο $\mathcal{I}_{\text{REQ}}$ παράγει προτάσεις τύπου 2.3 αναλύοντας τις προτάσεις $\mathcal{C}_1, \mathcal{C}_2$ με τύπο 3.3 και 2.3, ή 4.1 και 2.2, ή με τύπο 4.2 και 2.1.

Αρχικά υποθέτουμε ότι η πρόταση $\mathcal{C}_1$ είναι τύπου 3.3 ($B(x') \leftarrow C(x') \wedge [\mathbf{A}'(x')]$) και η $\mathcal{C}_2$ τύπου 2.3 ($C(f(x)) \leftarrow A(x)$), ενώ η $\mathcal{C}$ παράγεται αναλύοντας αυτές με μκε $\sigma_1 = \{x'/f(x)\}$. Προφανώς, το $\mathcal{Q}_i$ αναλύεται με $\mathcal{C}_1$ και παράγει την πρόταση:

$$\mathcal{Q}' = Q_P(\vec{s})\theta \leftarrow \bigwedge D_j(\vec{t_j})\theta \wedge C(x')\theta \wedge [\mathbf{A}'(x')\theta]$$

όπου θ είναι ένας μκε για τα $\{B(v), B(x')\}$. Όπως και στα προηγούμενα, μπορούμε να υποθέσουμε ότι θ είναι ένας μκε $\{x' \mapsto v\}$. Επιπλέον, αφού η x' δεν εμφανίζεται στο $\mathcal{Q}'$, η πρόταση $\mathcal{Q}'$ στην πραγματικότητα είναι της μορφής $\mathcal{Q}' = Q_P(\vec{s}) \leftarrow \bigwedge D_j(\vec{t_j}) \wedge C(v) \wedge [\mathbf{A}'(v)]$.

Επομένως, αφού οι μεταβλητές που εμφανίζονται στα $C(v), A'(v)$ είναι οι ίδιες με εκείνες στο $B(v)$, η $\mathcal{Q}'$ αναλύεται με την $\mathcal{C}_2$ με μκε σ, για να παραχθεί η πρόταση $\mathcal{Q}_{i+1}$. Επομένως, υπάρχουν παραγωγές της μορφής $\mathcal{Q}_i, \mathcal{C}_1 \vdash \mathcal{Q}'$ και $\mathcal{Q}', \mathcal{C}_2 \vdash \mathcal{Q}_{i+1}$. Επιπλέον, είναι ξεκάθαρο ότι $\mathcal{T} \vdash^{\mathcal{I}_{\text{REQ}}}_{j-1} \mathcal{C}_1$ και $\mathcal{T} \vdash^{\mathcal{I}_{\text{REQ}}}_{j-1} \mathcal{C}_2$.

Εν συνεχεία, υποθέτουμε ότι η $\mathcal{C}_1$ είναι τύπου 4.1 ($B(x') \leftarrow R(x', y)$) και η $\mathcal{C}_2$ τύπου 2.2 ($R(f(x), x) \leftarrow A(x)$), ενώ παράγεται χρησιμοποιώντας το μκε $\sigma_1 = \{x'/f(x), y \mapsto x\}$, η περίπτωση της 4.2 και 2.1 είναι παρόμοια. Ξανά, η $\mathcal{Q}_i$ αναλύεται με $\mathcal{C}_1$ και προκύπτει η πρόταση:

$$\mathcal{Q}' = Q_P(\vec{s})\theta \leftarrow \bigwedge D_j(\vec{t_j})\theta \wedge R(x', y)\theta$$

όπου θ είναι ο μκε για τα $\{B(v), B(x')\}$ και θ είναι ο μκε $\{x' \mapsto v\}$, επομένως, έχουμε $\mathcal{Q}' = Q_P(\vec{s}) \leftarrow \bigwedge D_j(\vec{t_j}) \wedge R(v, y)$. Εφόσον η v εμφανίζεται στο $B(v)$ της $\mathcal{Q}_i$ και η y είναι νέα στην $\mathcal{Q}'$, η $\mathcal{Q}'$ αναλύεται με την $\mathcal{C}_2$ με μκε $\sigma' = \sigma \cup \{y \mapsto x\}$ που παράγει την πρόταση $\mathcal{Q}_{i+1}$. Έτσι, υπάρχει και πάλι μια παραγωγή της μορφής $\mathcal{Q}_i, \mathcal{C}_1 \vdash \mathcal{Q}'$, $\mathcal{Q}', \mathcal{C}_2 \vdash \mathcal{Q}_{i+1}$, ενώ αφού ο $\mathcal{I}_{\text{REQ}}$ δεν παράγει ποτέ προτάσεις τύπου 4.1, $\mathcal{C}_1 \in \mathcal{T}$, και τελικά $\mathcal{T} \vdash^{\mathcal{I}_{\text{REQ}}}_{j-1} \mathcal{C}_2$, που είναι το ζητούμενο.

4. $\mathcal{C}$ έχει τύπο 3.3. Τότε, $\mathcal{Q}_i, \mathcal{C} \vdash^{\mathcal{I}_{\text{SLD}}} \mathcal{Q}_{i+1}$ έχει την παρακάτω μορφή:

$$\frac{Q_P(\vec{s}) \leftarrow \bigwedge D_j(\vec{t_j}) \wedge A(v) \quad A(x) \leftarrow B(x)}{Q_P(\vec{s})\sigma \leftarrow \bigwedge D_j(\vec{t_j})\sigma \wedge B(x)\sigma}$$

όπου σ είναι ένας μκε για το $\{A(v), A(x)\}$. Επιπλέον, ο $\mathcal{I}_{\text{REQ}}$ παράγει προτάσεις με τύπο 3.3 αναλύοντας προτάσεις $\mathcal{C}_1, \mathcal{C}_2$ είτε με τύπο 4.1 και 2.1, είτε με τύπο 3.2 και 2.2. Υποθέτουμε ότι η $\mathcal{C}_1$ έχει τύπο 4.1 ($A(x) \leftarrow R(x, y)$) και η $\mathcal{C}_2$ έχει τύπο 2.1 ($R(x, f(x)) \leftarrow B(x)$) ενώ η $\mathcal{C}$ παράγεται χρησιμοποιώντας το μκε

$\sigma_1 = \{y/f(x)\}$ (η περίπτωση των 4.2 και 2.2 είναι παρόμοια). Προφανώς, το $\mathcal{Q}$ αναλύεται με την $\mathcal{C}_1$ για να προκύψει η πρόταση:

$$\mathcal{Q}' = Q_P(\vec{s})\theta \leftarrow \bigwedge D_j(\vec{t_j})\theta \wedge R(x,y)\theta$$

Ξανά υποθέτουμε ότι η θ είναι ο μκε $\{x \mapsto v\}$ και το $\mathcal{Q}'$ είναι στην πραγματικότητα της μορφής $\mathcal{Q}' = Q_P(\vec{s}) \leftarrow \bigwedge D_j(\vec{t_j}) \wedge R(v,y)$. Έτισ, αφού η v εμφανίζεται στα άτομα $R(v,y)$ και $A(v)$ ενώ η y δεν εμφανίζεται στο $\mathcal{Q}_i$, η πρόταση $\mathcal{Q}'$ αναλύεται με $\mathcal{C}_2$ με μκε σ προκειμένου να προκύψει η $\mathcal{Q}_{i+1}$.

Ο παραπάνω ισχυρισμός υπονοεί ότι για μια πρόταση $\mathcal{C}$ τέτοια ώστε $\mathcal{T} \vdash^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{C}$ ο κανόνας συμπερασμού $\mathcal{Q}_i, \mathcal{C} \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}_{i+1}$ μπορεί να μετατραπεί σε μία ακολουθία από κανόνες της μορφής $\mathcal{Q}_i, \mathcal{C}_1 \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}'_1$, $\mathcal{Q}'_1, \mathcal{C}_2 \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}'_2, \ldots, \mathcal{Q}'_{n-1}, \mathcal{C}_n \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}_{i+1}$, τέτοια ώστε για όλα τα $1 \leq i \leq n$ ισχύει ότι $\mathcal{C}_i \in \mathcal{T}$. Έτσι, $\mathcal{T}, \mathcal{Q}_i \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}_{i+1}$ και σε συνδυασμό με την επαγωγική υπόθεση ($\mathcal{T}, \mathcal{Q} \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}_i$) προκύπτει το ζητούμενο $\mathcal{T}, \mathcal{Q} \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}_{i+1}$. □

Έπειτα, δείχνουμε ότι κάθε SLD παραγωγή μπορεί να μετατραπεί σε μία παραγωγή με βάση το $\mathcal{I}_{lite}$. Εφόσον η εκτύλιξη αντιστοιχεί στην κλασική δυαδική επίλυση, το τμήμα της παραγωγής με βάση το $\mathcal{I}_{lite}$ που δεν είναι τετριμμένο και πρέπει να αναλυθεί σχετίζεται με την εφαρμογή των κανόνων συρρίκνωσης. Όπως αναφέρθηκε ένας κανόνας συρρίκνωσης $\mathcal{Q}, \mathcal{C}_1, [\mathcal{C}_2] \vdash \mathcal{Q}'$ μπορεί να θεωρηθεί σαν περισσότερες εφαρμογές κανόνων της μορφής $\mathcal{Q}, \mathcal{C}_{i_1} \vdash \mathcal{Q}_1, \mathcal{Q}_1, \mathcal{C}_{i_2} \vdash \mathcal{Q}_2, \ldots, \mathcal{Q}_{n-1}, \mathcal{C}_{i_n} \vdash \mathcal{Q}'$ για $i_j \in \{0,1\}$, όπου $\mathcal{Q}, \mathcal{Q}'$ είναι προτάσεις χωρίς συνάρτηση και όλα τα $\mathcal{Q}_k$, $1 \leq k \leq n$ και $\mathcal{C}_1, \mathcal{C}_2$ αναφέρουν το *ίδιο* σύμβολο συνάρτησης f. Επομένως, διαισθητικά, οι παραγωγές από τον $\mathcal{I}_{lite}$ είναι κατά μία έννοια "συμπαγείς" ως προς τα συναρτησιακά σύμβολα, δηλαδή μία πρόταση που περιλαμβάνει ένα σύμβολο συνάρτησης ($\mathcal{Q}_1$) δημιουργείται από μία πρόταση που δεν περιλαμβάνει σύμβολο συνάρτησης ($\mathcal{Q}$) και οι κανόνες επίλυσης που ακολουθούν εφαρμόζουν δευτερεύουσες προτάσεις που επίσης αναφέρουν το ίδιο σύμβολο συνάρτησης επιχειρώντας να το απομακρύνουν από την πρόταση που παράγεται. Ονομάζουμε τις παραγωγές αυτές συμπαγείς ως προς τη συνάρτηση.

Ορισμός 3.2.2. Έστω $\Upsilon_1, \ldots, \Upsilon_n$ μία SLD παραγωγή όπου $\Upsilon_i, \mathcal{C}_i \vdash \Upsilon_{i+1}$ για $1 \leq i < n$. Υποθέτουμε ότι όλες οι $\Upsilon_i, 2 \leq i \leq n-1$ περιλαμβάνουν έναν όρο της μορφής $f(s_i)$ του ιδίου βάθους, και ότι οι $\mathcal{Q}_1$ και $\mathcal{Q}_n$ δεν περιλαμβάνουν συναρτήσεις στο σώμα τους, ή είναι προτάσεις με ισότητα που δε διαθέτουν συναρτήσεις στο σώμα. Λέμε ότι η παραγωγή είναι *συμπαγής ως προς τη συνάρτηση* (*function-compact*) εαν όλες οι δευτερεύουσες προκείμενες προτάσεις $\mathcal{C}_i$, $1 \leq i < n$, περιλαμβάνουν επίσης έναν όρο $f(x_i)$. △

Έτσι, προκειμένου να αποδείξουμε την ορθότητα του αλγορίθμου δείχνουμε ότι μια SLD παραγωγή που χρησιμοποιεί προτάσεις Horn σα δευτερεύουσες προτάσεις μπορεί να μετατραπεί σε μία άλλη που είναι συμπαγής ως προς τη συνάρτηση. Δείχνουμε αρχικά, ότι για κάθε παραγωγή $\mathcal{Q}_1, \ldots, \mathcal{Q}_n$, που δεν είναι συμπαγής ως προς τη συνάρτηση, υπάρχει ένας κανόνας με δευτερεύουσα προτάση κάποια $\mathcal{C}_k$ που δεν αναφέρει την f και μπορεί να μεταφερθεί "έξω" από την ακολουθία της παραγωγής,

και συγκεκριμένα είτε στην αρχή, οπότε και θα έχουμε $\mathcal{Q}_1, \mathcal{C}_k \vdash \mathcal{Q}'_2$, είτε στο τέλος, οπότε θα έχουμε $\mathcal{Q}_{n-1}, \mathcal{C}_k \vdash \mathcal{Q}_n$. Στην τελευταία περίπτωση αφού το $\mathcal{Q}_1$ είναι χωρίς συναρτήσεις και η $\mathcal{C}_k$ δεν αναφέρει το f, τότε το $\mathcal{Q}'_2$ πρέπει να μη διαθέτει συναρτήσεις. Στην δεύτερη περίπτωση αφού το $\mathcal{Q}_n$ δεν διαθέτει συναρτήσεις και η $\mathcal{C}_k$ δεν αναφέρει το f, τότε η $\mathcal{Q}'_{n-1}$ πρέπει να μη διαθέτει συναρτήσεις. Έτσι, αυτή η αναδιάταξη πρέπει είτε να σπρώχνει προς τα κάτω (στην πρώτη περίπτωση) είτε προς τα πάνω (στην δεύτερη περίπτωση) όλους τους κανόνες με δευτερεύουσες προτάσεις που αναφέρουν το f. Με την επαναλαμβανόμενη εφαρμογή αυτής της αναδιάταξης μεταξύ της πρώτης πρότασης όπου πρωτοεμφανίστηκε η f, μέχρι την τελευταία που την αναφέρει, θα εμφανίζονται μόνο δευτερεύουσες προκείμενες που αναφέρουν την f. Πρώτα παρουσιάζουμε ένα περισσότερο γενικό αποτέλεσμα που δεν βασίζεται στη DL-Lite και που μπορεί επίσης να χρησιμοποιηθεί αργότερα όταν επεκτείνουμε το λογισμό και για τις πιο εκφραστικές γλώσσες.

Λήμμα 3.2.3. *Κάθε παραγωγή που προκύπτει από κανόνες επίλυσης που έχουν ως δευτερεύουσες προκείμενες Horn-προτάσεις που διαθέτουν συναρτήσεις μόνο στην κεφαλή, μπορεί να μετατραπεί σε μία άλλη που είναι συμπαγής ως προς τη συνάρτηση.*

Απόδειξη. Θεωρούμε μία παραγωγή $\mathcal{Q}_1, \mathcal{Q}_2, \ldots, \mathcal{Q}_n$ από τον $\mathcal{I}_{\mathsf{SLD}}$ τέτοια ώστε όλες οι προτάσεις $\mathcal{Q}_2, \ldots, \mathcal{Q}_{n-1}$ περιλαμβάνουν έναν όρο που αναφέρει ένα σύμβολο συνάρτησης f ενώ οι $\mathcal{Q}_1$ και $\mathcal{Q}_n$ δεν αναφέρουν το f. Στα επόμενα αναφερόμαστε για ευκολία σε τέτοιες παραγωγές με $\langle Q_1, \mathcal{C}_1\rangle, \langle \mathcal{Q}_2, \mathcal{C}_2\rangle, \ldots, \langle \mathcal{Q}_n, \mathcal{C}_n\rangle$, όπου $\mathcal{Q}_i, \mathcal{C}_i \vdash \mathcal{Q}_{i+1}$.

Το λήμμα προκύπτει από τον ακόλουθο ισχυρισμό:

> *Ισχυρισμός 2:* Για κάθε παραγωγή όπως η παραπάνω και για την οποία υπάρχουν $\mathcal{Q}_j$ και $\mathcal{C}_j$ τέτοια ώστε η $\mathcal{C}_j$ δεν αναφέρει το f, τότε υπάρχει επίσης κάποια πρόταση $\mathcal{C}_k$ που δεν αναφέρει το f και είναι τέτοια ώστε οποιαδήποτε από τα παρακάτω είναι έγκυρες παραγωγές από το $\mathcal{I}_{\mathsf{SLD}}$:
>
> 1. $\langle \mathcal{Q}_1, \mathcal{C}_k\rangle, \langle \mathcal{Q}'_2, \mathcal{C}_1\rangle, \ldots, \langle \mathcal{Q}'_k, \mathcal{C}_{k-1}\rangle,$
> $\langle \mathcal{Q}_{k+1}, \mathcal{C}_{k+1}\rangle, \ldots, \langle \mathcal{Q}_n, \mathcal{C}_n\rangle$ ή
> 2. $\langle \mathcal{Q}_1, \mathcal{C}_1\rangle, \langle \mathcal{Q}_2, \mathcal{C}_2\rangle, \ldots, \langle \mathcal{Q}_k, \mathcal{C}_{k+1}\rangle,$
> $\langle \mathcal{Q}'_{k+1}, \mathcal{C}_{k+2}\rangle, \ldots, \langle \mathcal{Q}'_{n-1}, \mathcal{C}_k\rangle, \langle \mathcal{Q}_n, \mathcal{C}_n\rangle$

Σύμφωνα με τον ισχυρισμό κάποιο κανόνας με δευτερεύσουσα πρόταση $\mathcal{C}_k$ που δεν αναφέρει το f μπορεί να μεταφερθεί "έξω" από το σχετικό τμήμα της παραγωγής. Στην Περίπτωση 1., αφού το $\mathcal{Q}_1$ και η $\mathcal{C}_k$ δεν αναφέρουν το f, το ίδιο συμβαίνει και με το $\mathcal{Q}'_2$. Έτσι, η πρώτη πρόταση που αναφέρει το f είναι η $\mathcal{Q}'_3$. Στην Περίπτωση 2., και πάλι αφού οι $\mathcal{Q}_n, \mathcal{C}_k$ δεν αναφέρουν το f, το ίδιο συμβαίνει και με την $\mathcal{Q}_{n-1}$ (μία πρόταση χωρίς συναρτήσεις δεν μπορεί να δημιουργηθεί από έναν κανόνα με κύρια πρόταση που περιλαμβάνει συνάρτηση και μία δευτερεύσουσα που δεν περιλαμβάνει τη συνάρτηση). Συνεπώς, από επαναλαμβανόμενες εφαρμογές του παραπάνω μετασχηματισμού μπορούμε να αφαιρέσουμε όλους τους κανόνες που έχουν δευτερεύουσα πρόταση που δεν αναφέρει την f ώστε να προκύψει νέα "πρώτη" και "τελευταία" πρόταση που δεν περιέχουν συνάρτηση και είναι πιο κοντά η μία στην

άλλη. Η μετατροπή ξεκάθαρα τερματίζει και μπορεί να συμβεί σε όλα τα τμήματα μιας παραγωγής SLD. Έτσι, έπειτα από ένα πεπερασμένο αριθμό βημάτων αποκτούμε μία παραγωγή συμπαγή ως προς τη συνάρτηση.

Σε αυτό το σημείο δείχνουμε τον *Ισχυρισμό 2*.

Πρώτα, δείχνουμε ότι αν ένα ζεύγος από κανόνες $\mathcal{Q}_1, \mathcal{C}_1 \vdash \mathcal{Q}_2$ και $\mathcal{Q}_2, \mathcal{C}_2 \vdash \mathcal{Q}_3$ μπορεί να επαναγραφεί ως $\mathcal{Q}_1, \mathcal{C}_2 \vdash \mathcal{Q}'$ και $\mathcal{Q}', \mathcal{C}_1 \vdash \mathcal{Q}''$, τότε το $\mathcal{Q}''$ είναι στην πραγματικότητα $\mathcal{Q}_3$, δηλαδή αν αναδιατάξουμε το ζεύγος, τότε προκύπτει η ίδια τελική πρόταση. Έτσι, στον Ισχυρισμό 2, αφού η πρόταση $\mathcal{C}_k$ έχει μεταφερθει στην αρχή (τέλος), έχουμε πράγματι ότι $\mathcal{Q}'_k, \mathcal{C}_{k-1} \vdash \mathcal{Q}_{k+1}$ ($\mathcal{Q}'_{n-1}, \mathcal{C}_k \vdash \mathcal{Q}_n$).

Με περισσότερη λεπτομέρεια, ο κανόνας $\mathcal{Q}_1, \mathcal{C}_1 \vdash \mathcal{Q}_2$ είναι της μορφής:

$$\frac{Q_P(\vec{s}) \leftarrow \bigwedge D_j(\vec{t_j}) \wedge A \wedge B \quad A_1 \leftarrow A_2}{Q_P(\vec{s})\sigma \leftarrow \bigwedge D_j(\vec{t_j})\sigma \wedge A_2\sigma \wedge B\sigma}$$

όπου σ είναι ο μκε για τα $\{A, A_1\}$, ενώ ο $\mathcal{Q}_2, \mathcal{C}_2 \vdash \mathcal{Q}_3$ έχει τη μορφή:

$$\frac{Q_P(\vec{s})\sigma \leftarrow \bigwedge D_j(\vec{t_j})\sigma \wedge A_2\sigma \wedge B\sigma \quad B_1 \leftarrow B_2}{Q_P(\vec{s})\sigma\theta \leftarrow \bigwedge D_j(\vec{t_j})\sigma\theta \wedge A_2\sigma\theta \wedge B_2\theta}$$

όπου θ είναι ο μκε για τα $\{B\sigma, B_1\}$. Μετά την αναδιάταξη, ο κανόνας $\mathcal{Q}_1, \mathcal{C}_2 \vdash \mathcal{Q}'$ είναι της μορφής:

$$\frac{Q_P(\vec{s}) \leftarrow \bigwedge D_j(\vec{t_j}) \wedge A \wedge B \quad B_1 \leftarrow B_2}{Q_P(\vec{s})\theta' \leftarrow \bigwedge D_j(\vec{t_j})\theta' \wedge A\theta' \wedge B_2\theta'}$$

όπου θ' ένας μκε για τα $\{B, B_1\}$, ενώ ο $\mathcal{Q}', \mathcal{C}_2 \vdash \mathcal{Q}''$ έχει τη μορφή:

$$\frac{Q_P(\vec{s})\theta' \leftarrow \bigwedge D_j(\vec{t_j})\theta' \wedge A\theta' \wedge B_2\theta' \quad A_1 \leftarrow A_2}{Q_P(\vec{s})\theta'\sigma' \leftarrow \bigwedge D_j(\vec{t_j})\theta'\sigma' \wedge A_2\sigma' \wedge B_2\theta'\sigma'}$$

με σ' ένας μκε για τα $\{A_1, A\theta'\}$. Η πρόταση $\mathcal{Q}''$ περιλαμβάνει τα ίδια κατηγορήματαόπως το $\mathcal{Q}_3$. Για να δείξουμε ότι το $\mathcal{Q}''$ είναι ισοδύναμο (λαμβάνοντας υπόψη τις μετονομασίες μεταβλητών στο $\mathcal{Q}_3$) χρειάζεται να δείξουμε ότι οι ενοποιητές θ' και σ' υπάρχουν ώστε ο $\theta' \circ \sigma'$ αντιστοιχίζει μεταβλητές του $\mathcal{Q}$ με τον ίδιο τρόπο όπως και ο $\sigma \circ \theta$. Σε αυτή την κατέυθυνση, δεδομένης μία αντιστοίχισης $s_1 \mapsto s_2 \in \sigma$ και μίας άλλης $t_1 \mapsto t_2 \in \theta$, δείχνουμε πώς μπορούμε να κατασκευάσουμε αντιστοιχήσεις $s'_1 \mapsto s'_2 \in \sigma'$ και $t'_1 \mapsto t'_2 \in \theta'$ τέτοιες ώστε με βάση τις μεταβλητές που αντιστοιχίζονται, η εφαρμογή της σ και έπειτα της θ έχει το ίδιο αποτέλεσμα με την εφαρμογή πρώτα της θ' και έπειτα της σ'. Συνεπώς, αφού αυτό μπορεί να συμβεί για κάθε συνδυασμό αντιστοιχίσεων της σ και θ, η εφαρμογή της θ' και έπετια της σ' θα έχει το ίδιο αποτέλεσμα σε όλες τις μεταβλητές του ερωτήματος. Εφόσον υπάρχουν ενοποιητές υπάρχουν επίσης και μκε και θα έχουμε ότι $\mathcal{Q}_3 = \mathcal{Q}''$. Οι ενοποιητές θ' και σ' κατασκευάζονται σύμφωνα με τα σ και θ όπως προκύπτει, όπου τα 1. και 2. εφαρμόζονται εξαντλητικά και έπειτα τα 3., 4., 5., και 6. με τη σειρά:

1. Αν ο σ περιλαμβάνει την αντιστοίχιση $x \mapsto s$ και ο θ περιλαμβάνει την $w \mapsto s$, τότε πρόσθεσε την $w \mapsto x$ στο θ' και την $x \mapsto s$ στο σ'. Είναι προφανές ότι ο $\theta' \circ \sigma'$ αντιστοιχίζει τις x και w στους ίδους όρους με τον $\sigma \circ \theta$.

2. Αν ο σ περιλαμβάνει την $x \mapsto s$ και ο θ περιλαμβάνει το $w \mapsto g(s)$ για κάποια συνάρτηση g, τότε πρόσθεσε την $w \mapsto g(x)$ στο θ' και την $x \mapsto s$ στο σ'. Όπως και προηγουμένως, ο $\sigma \circ \theta$ αντιστοιχίζει τις x και w στους ίδιους όρους όπως $\sigma \circ \theta$.

3. Σε όλες τις υπόλοιπες περιπτώσεις αντέγραψε τις αντιστοιχίσεις του θ στο θ'.

4. Έστω ότι ο ενοιποιητής σ περιλαμβάνει την αντιστοίχιση $x \mapsto y$ και έστω ότι η x είναι μεταβλητή του $\mathcal{Q}_1$. Αν η θ' περιλαμβάνει μία αντιστοίχιση της μορφής $y \mapsto s$, τότε πρόσθεσε την $x \mapsto y$ στο σ'; διαφορετικά, πρόσθεσε την $x \mapsto s$ στο σ'; ομοίως αν $x \mapsto f(y) \in \sigma$ και $y \mapsto s \in \theta$, τότε πρόσθεσε την $x \mapsto f(s)$ στο σ'. Όπως και πριν, ο $\theta' \circ \sigma'$ αντιστοιχίζει μεταβλητές x και y στους ίδιους όρους όπως και ο $\sigma \circ \theta$.

5. Σε όλες τις άλλες περιπτώσεις αντέγραψε τις αντιστοιχίσεις του σ στο σ'.

6. Αν ο σ περιλαμβάνει την αντιστοίχιση $x \mapsto s$ και x είναι μεταβλητή του $\mathcal{C}_1$, τότε πρόσθεσε το $x \mapsto s\theta'$ στο σ'. Σε αυτή την περίπτωση η αντιστοίχιση δεν επηρεάζει τις μεταβλητές του $\mathcal{Q}''$ αφού το x ανήκει στην $\mathcal{C}_1$.

Σημειώνουμε ότι η κατασκευή τον ενοποιητών είναι καλώς ορισμένη: οι περιπτώσεις 1. και 2. είναι ανεξάρτητες η μία από την άλλη και δεν εξαρτώνται από αντιστοιχήσεις που εισήχθησαν νωρίτερα. Επιπλέον, η περίπτωση 3. είναι επίσης ανεξάρτητη. Η περίπτωση 4. έχει μία συνθήκη που αφορά την θ'. Παρολαυτά, η θ' δεν αλλάζει ξανά και έτσι η συνθήκη είτε ικανοποιείται είτε όχι και η κατασκευή του σ' είναι καώς ορισμένη. Τέλος, οι περιπτώσεις 5. και 6. είναι επίσης καλώς ορισμένες.

Πριν προχωρήσουμε επισημαίνουμε ότι ένα ζεύγος κανόνων συμπερασμού $\mathcal{Q}_1, \mathcal{C}_1 \vdash \mathcal{Q}_2$ και $\mathcal{Q}_2, \mathcal{C}_2 \vdash \mathcal{Q}_3$ δεν μπορεί να επαναγραφεί ως $\mathcal{Q}_1, \mathcal{C}_2 \vdash \mathcal{Q}'$ και $\mathcal{Q}', \mathcal{C}_1 \vdash \mathcal{Q}''$ παραμόνο αν η κεφαλή του $\mathcal{C}_2$ αναλύεται με κάποιο άτομο που εισάχθηκε από την επίλυση του $\mathcal{Q}_1$ με την πρόταση $\mathcal{C}_1$, δηλαδή, η $\mathcal{C}_1$ είναι της μορφής $A \leftarrow \mathbf{B}$, η $\mathcal{C}_2$ της μορφής $B \leftarrow \mathbf{C}$ και $B \in \mathbf{B}$ και έτσι, ο κανόνας με δευτερεύουσα πρόταση την $\mathcal{C}_2$ απαιτεί ότι θα προηγηθεί ο κανόνας με δευτερεύουσα την $\mathcal{C}_1$. Αν η αναδιάταξη των προηγούμενων κανόνων είναι επιτρεπτή, τότε λέμε ότι η $\mathcal{C}_2$ είναι *ανεξάρτητη* από την $\mathcal{C}_1$, ενώ στην αντίθετη περίπτωση λέμε ότι η $\mathcal{C}_2$ είναι *εξαρτώμενη*.

Τέλος, δείχνουμε ότι οι προτάσεις που δεν αναφέρουν ένα συναρτησιακό σύμβολο μπορούν πράγματι να μεταφερθούν είτε στην αρχή είτε στο τέλος της ακολουθίας.

Θεωρούμε μία ακολουθία $\mathcal{Q}_1, \ldots, \mathcal{Q}_n$ όπου οι $\mathcal{Q}_2, \ldots, \mathcal{Q}_{n-1}$ αναφέρουν ένα σύμβολο συνάρτησης f ενώ τα $\mathcal{Q}_1$ και $\mathcal{Q}_n$ δεν περιέχουν συναρτήσεις. Υποθέτουμε επίσης ότι καμία από τις προτάσεις $\mathcal{C}_i$ με $2 \leq i \leq n-1$ που δεν περιλαμβάνει την f δεν μπορεί να μεταφερθεί στην αρχή ή το τέλος.

Έστω ότι η προταση $\mathcal{C}_k$ είναι η πρώτη κύρια πρόταση που δεν αναφέρει το f. Από την υπόθεση δεν μπορεί να μετακινηθεί στο ξεκίνημα της ακολουθίας. Έτσι, η $\mathcal{C}_k$ έχει τη μορφή $A \leftarrow \mathbf{B}$ και υπάρχει κάποιος προηγούμενος κανόνας $\mathcal{Q}_j, \mathcal{C}_j \vdash \mathcal{Q}_{j+1}$ με $j > 1$, όπου $\mathcal{C}_j$ είναι της μορφής $F \leftarrow \mathbf{A}$ με $A \in \mathbf{A}$ οπότε και η $\mathcal{C}_k$ δεν μπορεί να μεταφερθεί εκτός της ακολουθίας. Αν η πρόταση $\mathcal{C}_j$ δεν περιλαμβάνει συναρτήσεις, τότε η υπόθεση βρίσκει εφαρμογή και έτσι υπάρχει επίσης κάποια άλλη πρόταση από

την οποία εξαρτάται η $\mathcal{C}_j$. Επομένως, μπορεί να προκύψει ότι για κάποια πρόταση που δεν μπορεί να αναδιαταχθεί υπάρχει κάποια άλλη που δεν επιτρέπει την αναδιάταξη, η οποία έχει το f στην κεφαλή. Έστω ότι η πρόταση αυτή είναι η $\mathcal{C}_j$, δηλαδή για την $\mathcal{C}_j$ γνωρίζουμε ότι το F περιλαμβάνει το f. Προκύπτει επίσης, ότι καμία ενδιάμεση πρόταση $\mathcal{C}_\ell$, $j < \ell < k$, δεν μπορεί να εξαρτάται από την $\mathcal{C}_j$, διαφορετικά ο κανόνας με την $\mathcal{C}_\ell$ θα απομάκρυνε το A και έτσι, η $\mathcal{C}_k$ δε θα εξαρτόταν από την $\mathcal{C}_j$. Συνεπώς, όλες οι προτάσεις που βρίσκονται ενδιάμεσα και δεν εξαρτώνται απο την $\mathcal{C}_j$ μπορούν να μεταφερθούν πριν από την $\mathcal{C}_j$, δηλαδή μπορούν να αναδιαταχθούν ώστε $j = k - 1$.

Στη συνέχεια και πάλι από την υπόθεση η πρόταση $\mathcal{C}_k$ δεν μπορεί να μεταφερθεί προς τα κάτω. Έτσι, αυτό υπονοεί ότι υπάρχει επίσης κάποια πρόταση $\mathcal{C}_m$ που εξαρτάται από την $\mathcal{C}_k$, δηλαδή έχει τη μορφή $B \leftarrow \mathbf{C}$, όπου $B \in \mathbf{B}$. Και πάλι, καμία πρόταση μεταξύ των $\mathcal{C}_k$ και $\mathcal{C}_m$ δεν μπορεί να εξαρτάται από την $\mathcal{C}_k$. Επιπλέον, δεν μπορούν να εξαρτώνται και από την $\mathcal{C}_j$ καθώς το A έχει απομακρυνθεί από τον κανόνα με δευτερεύουσα την $\mathcal{C}_k$. Έτσι, όλες αυτές οι προτάσεις μπορούν επίσης να μεταφερθούν πριν την $\mathcal{C}_j$, δηλαδή την $\mathcal{C}_{k-1}$ στην αναδιαταγμένη ακολουθία.

Εφαρμόζοντας επαναλαμβανόμενα αυτή τη λογική μπορούμε να αναδιατάξουμε την ακολουθία έτσι ώστε μετά την πρόταση $\mathcal{C}_{k-1}$ να έχουμε κανόνες με δευτερεύουσες προτάσεις τις $\mathcal{C}_{k-1}, \mathcal{C}_k, \mathcal{C}_{k+1}$ ώστε η κεφαλή της κάθε πρότασης να αναλύεται με κάποιο άτομο στο σώμα της πρότασης που βρίσκεται ακριβώς στα αριστερά.

Έστω η πρόταση $\mathcal{C}^{k+1}$ που παράγεται από τους κανόνες $\mathcal{C}_{k-1}, \mathcal{C}_k \vdash \mathcal{C}'$, $\mathcal{C}', \mathcal{C}_{k+1} \vdash \mathcal{C}^{k+1}$. Από τη μορφή των προτάσεων που χρησιμοποιούνται σε αυτούς τους κανόνες η $\mathcal{C}^{k+1}$ πρέπει να έχει τη μορφή $F \leftarrow \mathbf{C}$. Επιπλέον, γίνεται εύκολα αντιληπτό ότι αντί να παράγουμε το $\mathcal{Q}_{k+2}$ από την ακολουθία κανόνων $\mathcal{Q}_{k-1}, \mathcal{C}_{k-1} \vdash \mathcal{Q}_k$, $\mathcal{Q}_k, \mathcal{C}_k \vdash \mathcal{Q}_{k+1}$, και $\mathcal{Q}_{k+1}, \mathcal{C}_{k+1} \vdash \mathcal{Q}_{k+2}$ μπορούμε να το παράγουμε ως συμπέρασμα των $\mathcal{Q}_{k-1}$ και $\mathcal{C}^{k-1}$. Επίσης, εφόσον το $\mathcal{Q}_{k-1}$ αναφέρει την f, ο κανόνας $\mathcal{Q}_{k-1}, \mathcal{C}_{k-1} \vdash \mathcal{Q}_k$ παράγει μία πρόταση που επίσης αναφέρει την f και η πρόταση $\mathcal{C}^{k+1}$ έχει την ίδια κεφαλή με την $\mathcal{C}_{k-1}$, τότε και η $\mathcal{Q}_{k+2}$ θα περιλάμβανε το f.

Συμπερασματικά, αν η πρόταση $\mathcal{C}_{k+1}$ είναι η τελευταία της ακολουθίας, δηλαδή η $\mathcal{C}_{n-1}$, τότε καταλήγουμε σε αντίφαση: σε αυτή την περίπτωση το $\mathcal{Q}_{k+2}$ είναι το $\mathcal{Q}_n$ που όπως έχουμε δείξει πρέπει να περιλαμβάνει το f. Διαφορετικά, υπάρχουν άλλοι κανόνες ανάμεσα στις $\mathcal{C}_{k+1}$ και $\mathcal{C}_{n-1}$. Αρχικά, σημειώνουμε ότι οι δευτερεύουσες προτάσεις των κανόνων δεν μπορεί να εξαρτώνται από την $\mathcal{C}_{k+1}$. Αν υπάρχει κάποια πρόταση που είναι εξαρτώμενη, τότε η ακολουθία αναδιατάσσεται μεταφέροντας την μετά την $\mathcal{C}_{k+1}$, και υπολογίζεται η $\mathcal{C}^{k+2}$ όπου $\mathcal{C}^{k+1}, \mathcal{C}_{k+2} \vdash \mathcal{C}^{k+2}$; διαφορετικά, ξεκινώντας από την κορυφή όλες η ανεξάρτητες προτάσεις σπρώχνονται μία προς μία πριν την $\mathcal{C}_{k-1}$. Μπορούμε να επαναλάβουμε αυτή την αναδιάταξη μέχρις ότου η τελική ακολουθία να αποτελείται από κανόνες που έχουν δευτερεύουσες προτάσεις της μορφής $\mathcal{C}_{k-1}, \mathcal{C}_k, \mathcal{C}_{k+1}, \mathcal{C}_{k+2}, \ldots, \mathcal{C}_{n-1}$, όλες οι προτάσεις εξαρτώνται άμεσα από την πρόταση στα αριστερά και όπως αναφέρθηκε ήδη το $\mathcal{Q}_n$ μπορεί να παραχθεί από κανόνα υπερεπίλυσης του $\mathcal{Q}_{k-1}$ και κάποιας πρότασης $\mathcal{C}^{n-1}$ που κατασκευάζεται αναλύοντας μία προς μία με επαναληπτικό τρόπο όλες τις δευτερεύουσες προτάσεις της προηγούμενης λίστας. Επομένως, καταλήγουμε και πάλι σε αντίφαση.

□

Η επόμενη πρόταση προκύπτει απ' ευθείας από τη μορφή των προτάσεων στα σώματα ορολογίας σε DL-Lite.

Πρόταση 3.2.4. *Έστω $\mathcal{T}$ ένα σώμα ορολογίας σε DL-Lite και $\mathcal{Q}$ ένα ερώτημα. Κάθε παραγωγή από το $\mathcal{T} \cup \{\mathcal{Q}\}$ με $\mathcal{I}_{\mathsf{SLD}}$ χρησιμοποιεί ως δευτερεύουσες προτάσεις Horn-προτάσεις που έχουν όρους με συναρτήσεις μόνο στην κεφαλή.*

Συνοψίζοντας, μία παραγωγή από το $\mathcal{T} \cup \{\mathcal{Q}\}$ με βάση το $\mathcal{I}_{\mathsf{REQ}}$ μπορεί να μετατραπεί σε παραγωγή με βάση το $\mathcal{I}_{\mathsf{SLD}}$ η οποία μπορεί με τη σειρά της να μετασχηματιστεί σε μία συμπαγή ως προς τη συνάρτηση παραγωγή. Μπορούμε σε αυτό το σημείο να αποδείξουμε την ορθότητα του $\mathcal{I}_{lite}$.

Θεώρημα 3.2.5. *Έστω ένα σώμα ορολογίας σε DL-Lite $\mathcal{T}$ και ένα ΣΕ $\mathcal{Q}$. Κάθε παραγωγή από το $\mathcal{T} \cup \{\mathcal{Q}\}$ από τον $\mathcal{I}_{lite}$ τερματίζει. Επιπλέον, το* Rapid-Lite$(\mathcal{Q}, \mathcal{T})$ *είναι μία επαναγραφή του $\mathcal{Q}$ με βάση το $\mathcal{T}$.*

Απόδειξη. Ο τερματισμός είναι άμεση συνέπεια των παρακάτω:

- ο $\mathcal{I}_{lite}$ παράγει πάντοντε προτάσεις χωρίς συναρτήσεις.
- Εάν το $\mathcal{Q}'$ παράγεται από το $\mathcal{Q}$, τότε το $\mathcal{Q}'$ έχει το πολύ τον ίδιο αριθμό από δεσμευμένες μεταβλητές όπως και το $\mathcal{Q}$.

Συνεπώς, υπάρχει μόνο ένας πεπερασμένος αριθμός ερωτημάτων που μπορούν να παραχθούν από το $\mathcal{T}$ και $\mathcal{Q}$ χρησιμοποιώντας το λογισμό $\mathcal{I}_{lite}$ (λαμβάνοντας υπόψη τη μετονομασία των μεταβλητών).

Στα επόμενα αποδεικνύουμε την ορθότητα. Έστω $\mathcal{R}_Q =$ Rapid-Lite$(\mathcal{Q}, \mathcal{T})$. Προκειμένου να δείξουμε ότι το $\mathcal{R}_Q$ είναι μία επαναγραφή ΣΣΕ πρέπει να δείξουμε ότι για κάθε $\mathcal{A}$ έχουμε $\mathsf{cert}(\mathcal{Q}, \mathcal{T} \cup \mathcal{A}) = \mathsf{cert}(\mathcal{R}_Q, \mathcal{A})$.

Προκειμένου να δείξουμε ότι $\mathsf{cert}(\mathcal{Q}, \mathcal{T}, \mathcal{A}) \supseteq \mathsf{cert}(\mathcal{R}_Q, \mathcal{A})$, αρκεί να δείξουμε ότι $\mathcal{T} \cup \{\mathcal{Q}\} \models \mathcal{R}_Q$, που ισοδυναμεί με το $\mathcal{T} \cup \{\mathcal{Q}\} \models \mathcal{Q}_i$ για κάθε $\mathcal{Q}_i \in \mathcal{R}_Q$. Κάτι τέτοιο προκύπτει απευθείας αφού κάθε $\mathcal{Q}_i$ κατασκευάζεται χρησιμοποιώντας ένα λογισμό που βασίζεται στην επίλυση.

Στη συνέχεια, δείχνουμε ότι $\mathsf{cert}(\mathcal{Q}, \mathcal{T}, \mathcal{A}) \subseteq \mathsf{cert}(\mathcal{R}_Q, \mathcal{A})$. Θεωρούμε τον αλγόριθμο του Requiem. Αρκεί να δείξουμε ότι για κάθε $\mathcal{Q}' \in \mathsf{RQR}(\mathcal{Q}, \mathcal{T})$ ένα ερώτημα $\mathcal{Q}'' \in \mathcal{R}_Q$ υπάρχει τέτοιο ώστε τα $\mathcal{Q}'$ και $\mathcal{Q}''$ να είναι ισοδύναμα με βάση τη μετονομασία των μεταβλητών. Το $\mathsf{RQR}(\mathcal{Q}, \mathcal{T})$ κατασκευάζεται σε δύο βήματα. Αρχικά, οι κανόνες του $\mathcal{I}_{\mathsf{REQ}}$ εφαρμόζονται εξαντλητικά στο $\mathcal{T} \cup \{\mathcal{Q}\}$ για να αποκτήσουμε ένα (χωρίς αναδρομή) πρόγραμμα datalog P, και έπειτα το P εκτυλίσσεται για να αποκτήσουμε ένα ΣΣΕ.

Με βάση το Λήμμα 3.2.1, κάθε πρόταση τύπου 1 στο P μπορεί επίσης να προκύψει από το $\mathcal{T} \cup \{\mathcal{Q}\}$ με τον $\mathcal{I}_{\mathsf{SLD}}$, ενώ από το Λήμμα 3.2.3 η SLD παραγωγή μπορεί να θεωρηθεί ότι είναι συμπαγής ως προς τη συνάρτηση. Έτσι, αρκεί να δείξουμε ότι για κάθε πρόταση τύπου 1 που παράγεται από το $\mathcal{T} \cup \{\mathcal{Q}\}$ με βάση τον $\mathcal{I}_{\mathsf{SLD}}$ προκύπτει επίσης από $\mathcal{T} \cup \{\mathcal{Q}\}$ με βάση τον $\mathcal{I}_{lite}$.

Έστω $\mathcal{Q}_i$ το i-οστό ερώτημα που παράγεται από το $\mathcal{T} \cup \{\mathcal{Q}\}$ με $\mathcal{I}_{\mathsf{SLD}}$. Χρησιμοποιούμε επαγωγή.

Βασική περίπτωση (i=0): Στην περίπτωση που $\mathcal{Q}_0 = \mathcal{Q}$ και από τον ορισμό του Rapid-Lite$(\mathcal{Q}, \mathcal{T})$, έχουμε ότι $\mathcal{Q}_0 \in R_Q$.

Επαγωγικό βήμα: Υποθέτουμε ότι για όλα τα $1 \leq i \leq n$, όλες οι προτάσεις χωρίς συνάρτηση $\mathcal{Q}_i$ που προκύπτουν από τον $\mathcal{I}_{\text{SLD}}$ προκύπτουν επίσης από το $\mathcal{I}_{lite}$ και υποθέτουμε ότι μία νέα πρόταση χωρίς συνάρτηση $\mathcal{Q}_{i+1}$ προκύπτει έπειτα από το $\mathcal{I}_{\text{SLD}}$. Υπάρχουν δύο περιπτώσεις:

1. το $\mathcal{Q}_{i+1}$ προκύπτει από κάποιο $\mathcal{Q}_j$, $j \leq i$ χωρίς συναρτήσεις με δευτερεύουσα κάποια πρόταση $\mathcal{C}_j \in \mathcal{T}$. Εφόσον και τα δύο $\mathcal{Q}_j$ και $\mathcal{Q}_{i+1}$ δεν διαθέτουν συναρτήσεις ο εν λόγω κανόνας μπορεί να περιγραφεί από τον κανόνα της εκτύλιξης. Έτσι, $\mathcal{Q}_j, \mathcal{C}_j \vdash^{\mathcal{I}_{lite}} \mathcal{Q}_{i+1}$ και άρα $\mathcal{Q}_{i+1} \in \mathcal{R}_Q$.

2. το $\mathcal{Q}_{i+1}$ παράγεται από κάποιο $\mathcal{Q}_j$ με βάση το $\mathcal{I}_{\text{SLD}}$ με μία παραγωγή SLD της μορφής $\mathcal{Q}_j, \mathcal{Q}_{j+1}, \ldots, \mathcal{Q}_{j+k}$, με $\mathcal{Q}_{j+k} = \mathcal{Q}_{i+1}$, όπου όλες οι ενδιάμεσες προτάσεις περιλαμβάνουν έναν όρο $f(s_j)$, ενώ, επίσης, από την ιδιότητα της συμπαγούς ως προς τη συνάρτηση παραγωγής, όλες οι δευτερεύουσες προτάσεις περιλαμβάνουν τον όρο $f(t_j)$. Από τον ορισμό του $\mathcal{I}_{\text{SLD}}$, όλες οι δευτερεύουσες προτάσεις ανήκουν στο $\mathcal{T}$ που είναι κανονικοποιημένο. Επομένως αφού τα συναρτησιακά σύμβολα είναι μοναδικά ανά αξίωμα της μορφής $\exists R.A$, όλες οι δευτερεύουσες είναι της μορφής $R(x, f(x)) \leftarrow \mathbf{D}$ ή $A(f(x)) \leftarrow \mathbf{C}$, ενώ τα άτομα όλων των ενδιάμεσων προτάσεων τύπου 1 πάνω στα οποία αναλύονται είναι $R(u_j, f(s_j))$, $R(f(s_j), u_j)$, ή $A(f(s_j))$ όπου s_j, u_j είναι όροι. Επιπλέον, αφού το $\mathcal{Q}_j$ δε διαθέτει συνάρτηση και το $\mathcal{Q}_{j+1}$ αναφέρει την f, τότε το άτομο του $\mathcal{Q}_j$ που αναλύεται με την $\mathcal{C}_j$ είναι της μορφής $R(s,t)$, ή $A(t)$, και ο μκε περιλαμβάνει μια αντιστοίχιση της μορφής $x \mapsto f(y)$ για μία μεταβλητή x που εμφανίζεται δύο τουλάχιστον φορές στο σώμα του $\mathcal{Q}_j$ και δεν είναι μεταβλητή απάντησης (στο εξής, θα ονομάζουμε τις μεταβλητές αυτές δεσμευμένες).

   Έστω ότι τα S_1, S_2 περιλαμβάνουν όλα τα άτομα ρόλου και έννοιας αντίστοιχα του $\mathcal{Q}_j$ που συμμετέχουν σε έναν κανόνα. (Οποιοδήποτε από τα δύο σύνολα μπορεί να είναι κενό). Αφού υπάρχει ένας μκε για όλες τις ενδιάμεσες SLD παραγωγές, υπάρχει επίσης ένας άμεσος μκε για τα $S_1 \cup \{R(x, f(x))\}$ και $S_2 \cup \{A(f(x)\}$ [48]. Επιπλέον, όπως έχει αναλυθεί και στα παραπάνω από την κατασκευή του άμεσου μκε, θα πρέπει να περιλαμβάνει μία αντιστοίχιση της μορφής $x \mapsto f(y')$ όπου x είναι μία δεσμευμένη μεταβλητή. Συνεπώς, υπάρχει ένας κανόνας της μορφής $\mathcal{Q}_j, \mathcal{C}_1, [\mathcal{C}_2] \vdash \mathcal{Q}_{i+1}$ που περιγράφεται από τον κανόνα συρρίκνωσης.

Έτσι, σε κάθε περίπτωση όλες οι προτάσεις τύπου 1 προκύπτουν από τον $\mathcal{I}_{lite}$.

Τελικά, δείχνουμε ότι όλα τα ερωτήματα που παράγονται με εκτύλιξη παράγονται επίσης και από τον $\mathcal{I}_{lite}$. Κάτι τέτοιο αποδεικνύεται άμεσα, αφού οι κανόνες εκτύλιξης που εφαρμόζει το $\mathcal{I}_{\text{REQ}}$ μπορούν να ξεδιπλωθούν σε κανόνες που χρησιμοποιούν μόνο προτάσεις από το $\mathcal{T}$ που παράγουν πάντα προτάσεις τύπου 1 χωρίς συναρτήσεις. Έτσι, μπορούν να περιγραφούν από τον κανόνα εκτύλιξης του $\mathcal{I}_{lite}$. Συμπεραίνουμε ότι όλα τα ερωτήματα που παράγονται στο βήμα της εκτύλιξης ανήκουν και στο $\mathcal{R}_Q$. □

Κεφάλαιο 4

Επαναγραφή με επίλυση στην $\mathcal{ELHI}$

Στην ενότητα αυτή επεκτείνουμε το σύστημα κανόνων συμπερασμού $\mathcal{I}_{lite}$, με σκοπό να αναπτύξουμε έναν αλγόριθμο για οντολογίες σε $\mathcal{ELHI}$. Αξιοποιώντας τους κανόνες επίλυσης που ορίζονται στον $\mathcal{I}_{lite}$, ορίζεται ένας βελτιστοποιημένος αλγόριθμος για την περισσότερο εκφραστική $\mathcal{ELHI}$. Βασική διαφορά της γλώσσας σε σχέση με την DL-Lite, είναι ότι η $\mathcal{ELHI}$ επιτρέπει προτάσεις της μορφής $B(x) \leftarrow R(x,y) \wedge (x)$. Στο τελευταίο μέρος της ενότητας αποδεικνύουμε την ορθότητα του λογισμού.

4.1 Κανόνες επίλυσης του $\mathcal{I}_{\mathcal{EL}}$

Μία προφανής προσέγγιση προκειμένου να προκύψει ένας λογισμός επαναγραφής για την $\mathcal{ELHI}$ θα ήταν η επέκταση του Ορισμού 3.1.4 ώστε να επιτρέπει τυχαίες RA-προτάσεις, δηλαδή προτάσεις της μορφής $B(x) \leftarrow R(x,y) \wedge \mathbf{A}(x)$, ως δευτερεύουσες στον κανόνα της συρρίκνωσης και της εκτύλιξης.

Παράδειγμα 4.1.1. Έστω $\mathcal{T}$ ένα σώμα ορολογίας σε $\mathcal{ELHI}$ που αποτελείται από τις παρακάτω προτάσεις:

$$C(x) \leftarrow S(x,y) \wedge D(y) \tag{4.1}$$

$$S(f(x),x) \leftarrow B(x) \tag{4.2}$$

$$K(x) \leftarrow S(y,x) \wedge C(y) \tag{4.3}$$

και έστω το ερώτημα $\mathcal{Q}_1 = Q(x) \leftarrow K(x)$.

Από τον κανόνα της εκτύλιξης στις $\mathcal{Q}_1$ και (4.3) παράγεται το $\mathcal{Q}_2 = Q(x) \leftarrow S(y,x) \wedge C(y)$. Έπειτα, εφαρμόζοντας και πάλι εκτύλιξη στις $\mathcal{Q}_2$ και (4.1) παράγεται το $\mathcal{Q}_3 = Q(x) \leftarrow S(y,x) \wedge S(y,z) \wedge D(z)$. Τέλος ο κανόνας συρρίκνωσης στις $\mathcal{Q}_3$ και (4.2) παράγει $\mathcal{Q}_4 = Q(x) \leftarrow B(x) \wedge D(x)$. Προκύπτει ότι το σύνολο $\mathcal{R} = \{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3, \mathcal{Q}_4\}$ είναι επαναγραφή του $\mathcal{Q}$ με βάση το $\mathcal{T}$. $\diamondsuit$

Παρόλα αυτά, όπως φαίνεται και από το προηγούμενο παράδειγμα χρησιμοποιώντας τις RA-προτάσεις ως δευτερεύουσες προκείμενες, ενδέχεται να προκύψουν προτάσεις με περισσότερες μεταβλητές από όσες περιλαμβάνει η κύρια προκείμενη. Για

παράδειγμα, η πρόταση $\mathcal{Q}_3$ περιλαμβάνει μία μεταβλητή z που δεν εμφανίζεται στο $\mathcal{Q}_2$. Η μη ελεγχόμενη αύξηση του πλήθους των μεταβλητών στις παραγόμενες προτάσεις, έχει άμεση επίπτωση στη συνθήκη τερματισμού του αλγορίθμου. Στη γενική περίπτωση, θα μπορούσε κανείς να αντιμετωπίσει το πρόβλημα τερματισμού, ορίζοντας ένα ανώτατο όριο στον αριθμό των κανόνων που μπορούν να συμμετέχουν οι RA-προτάσεις ως δευτερεύουσες. Έτσι, καθορίζεται ένα ανώτατο φράγμα στον αριθμό των μεταβλητών που μπορούν να προκύψουν στη συμπερασματική πρόταση. Το πρόβλημα σχετίζεται με το πρόβλημα της δυνατότητας οριοθέτησης ερωτήματος και κατηγορήματος, το οποίο έχει μελετηθεί [23] θεωρητικά, δεν έχουν όμως σχεδιαστεί πρακτικοί αλγόριθμοι.

Συνεπώς, προκειμένου να αναπτυχθεί ένας αποδοτικός αλγόριθμος που να τερματίζει, ο λογισμός επαναγραφής θα πρέπει να αποκλύει τη χρήση RA-προτάσεων ως δευτερεύουσες προκείμενες κατά την εφαρμογή των κανόνων της επίλυσης. Προκειμένου όμως, ο λογισμός να καλύπτει την εκφραστικότητα της $\mathcal{ELHI}$, είναι απαραίτητο να κατασκευάζει τις ενδιάμεσες προτάσεις που μπορούν να προκύψουν από τις RA-προτάσεις. Το επόμενο παράδειγμα εξηγεί πως συμπεριφέρεται ο λογισμός $\mathcal{I}_{\mathsf{REQ}}$ στην είσοδο του Παραδείγματος 4.1.1.

Παράδειγμα 4.1.2. Θεωρούμε το σώμα ορολογίας $\mathcal{T}$ και το ερώτημα $\mathcal{Q}_1$ από το Παράδειγμα 4.1.1. Όταν ο αλγόριθμος Requiem εφαρμοστεί στα $\mathcal{T}$ και $\mathcal{Q}_1$ εκτελεί τους παρακάτω κανόνες:

$$(4.1), (4.2) \;\vdash\; C(f(x)) \leftarrow B(x) \wedge D(x) \tag{4.4}$$

$$(4.3), (4.2) \;\vdash\; K(x) \leftarrow B(x) \wedge C(f(x)) \tag{4.5}$$

$$(4.5), (4.4) \;\vdash\; K(x) \leftarrow B(x) \wedge D(x) \tag{4.6}$$

$$\mathcal{Q}_1, (4.6) \;\vdash\; Q(x) \leftarrow B(x) \wedge D(x) \tag{4.7}$$

Το σύνολο $\mathcal{R} = \{\mathcal{Q}_1, (4.1), (4.3), (4.7)\}$ είναι μία (datalog) επαναγραφή του $\mathcal{Q}_1$ με βάση το $\mathcal{T}$.

Αρχικά, παρατηρούμε ότι αν επεκτείνουμε τον κανόνα της συρρίκνωσης ώστε να επιτρέπει RA-προτάσεις ως κύριες, τότε οι κανόνες που εφαρμόζονται για να παράγουν την πρόταση (4.6) από την (4.3) μπορούν να αντικατασταθούν από ένα μόνο κανόνα συρρίκνωσης στην πρόταση (4.3) με δευτερεύσουσες τις (4.2) και (4.4). Παρόλα αυτά παρατηρούμε ότι η πρόταση (4.4) παράγεται αναλύοντας μια RA-πρόταση μαζί με μία DL-Lite-πρόταση, ενώ αυτή η αλληλεπίδραση δεν επιτρέπεται σε κάποιον από τους κανόνες του λογισμού $\mathcal{I}_{lite}$. ♢

Μελετώντας το παραπάνω παράδειγμα, ο λογισμός για τις $\mathcal{ELHI}$ οντολογίες που προτείνουμε αποτελείται πρώτον, από ένα νέο κανόνα συμπερασμού που μπορεί να παράγει προτάσεις όπως η (4.4), από RA-προτάσεις και μία πρόταση της μορφής (4.2), και δεύτερον, από μία επέκταση του κανόνα της εκτύλιξης και συρρίκνωσης που θα επιτρέπουν όχι μόνο κύριες προτάσεις με τύπο 1 αλλά και RA-προτάσεις ώστε να είναι δυνατή η παραγωγή για παράδειγμα της πρότασης (4.6) από τις (4.3), (4.2), και (4.4). Λόγω του νέου κανόνα, που μπορεί να παράγει προτάσεις με συναρτησιακό σύμβολο στην κεφαλή για παράδειγμα, η πρόταση (4.4), είναι δυνατό περισσότερες από δύο

προτάσεις να αναφέρουν το ίδιο σύμβολο συνάρτησης f. Επομένως, σε αντίθεση με την περίπτωση της DL-Lite, ένα συναρτησιακό σύμβολο είναι δυνατό να αντιστοιχεί σε περισσότερες από δύο προτάσεις. Για το λόγο αυτό ο κανόνας της συρρίκνωσης πρέπει να επεκταθεί ώστε να επιτρέπει ν δευτερεύουσες προτάσεις αντί για δύο το πολύ.

Ορισμός 4.1.3. Έστω Υ ένα ΣΕ ή μία RA-πρόταση, και $\mathcal{C}_{(i)}$ DL-Lite-προτάσεις. Με $\mathcal{I}_{\mathcal{EL}}$ συμβολίζουμε το λογισμό που αποτελείται από τους κανόνες του Πίνακα 4.1. Τέλος, για ένα ΣΕ $\mathcal{Q}$ και ένα σώμα ορολογίας $\mathcal{T}$ σε $\mathcal{ELHI}$, το Rapid-EL$(\mathcal{Q}, \mathcal{T})$ ορίζεται ως το σύνολο των προτάσεων χωρίς συναρτήσεις που προκύπτουν από το $\mathcal{T} \cup \{\mathcal{Q}\}$ με βάση το λογισμό $\mathcal{I}_{\mathcal{EL}}$. △

Πίνακας 4.1: Οι κανόνες του λογισμού $\mathcal{I}_{\mathcal{EL}}$

εκτύλιξη (*unfolding*):

$$\frac{\Upsilon \quad \mathcal{C}}{\Upsilon'\sigma}$$

όπου αν $x \mapsto f(y) \in \sigma$ τότε $x \in$ unbound(Υ).

συρρίκνωση (*shrinking*):

$$\frac{\Upsilon \quad \mathcal{C}_1 \; [\mathcal{C}_2 \; \ldots \; \mathcal{C}_n]}{\Upsilon'\sigma}$$

όπου $\Upsilon'\sigma$ είναι ένα συμπέρασμα χωρίς συναρτήσεις του Υ με τις $\mathcal{C}_1, \ldots, \mathcal{C}_n$, και υπάρχει κάποιο $x \mapsto f(y) \in \sigma$ ώστε $x \notin$ unbound(Υ).

κανόνας-συνάρτησης (*function-rule*):

$$\frac{B(x) \leftarrow R(x,y) \wedge [\mathbf{C}(y)] \quad R(f(x),x) \leftarrow A(x)}{B(f(x)) \leftarrow A(x) \wedge [\mathbf{C}(x)]}$$

$$\frac{B(x) \leftarrow R(y,x) \wedge [\mathbf{C}(y)] \quad R(x,f(x)) \leftarrow A(x)}{B(f(x)) \leftarrow A(x) \wedge [\mathbf{C}(x)]}$$

Σημειώνουμε ότι η επέκταση του κανόνα της συρρίκνωσης με ν δευτερεύσουσες προκείμενες, δεν αναμένεται να επηρεάσει την επίδοση του αλγορίθμου στην πράξη. Κάτι τέτοιο συμβαίνει γιατί ο κανόνας εφαρμόζεται με περισσότερες από δύο δευτερεύουσες, μόνο στην περίπτωση που ο κανόνας συνάρτησης έχει κατασκευάσει νέες προτάσεις τύπου 2.3 με κάποιο σύμβολο συνάρτησης στην κεφαλή. Την ίδια στιγμή, ο κανόνας συνάρτησης ανταποκρίνεται σε μία σύνθετη αλληλεπίδραση μεταξύ κάποιας πρότασης που περιλαμβάνει $R(x, f(x))$ $(R(f(x), x))$, με κάποια RA-πρόταση που περιλαμβάνει τον αντίστροφο $R(y, x)$ $(R(x, y))$, κάτι που στην πράξη δεν αναμένεται να συμβαίνει συχνά. Επίσης, σημειώνουμε ότι η εφαρμογή του λογισμού $\mathcal{I}_{\mathcal{EL}}$ σε κάποια είσοδο $\mathcal{T} \cup \{\mathcal{Q}\}$ μπορεί να διαμεριστεί σε δύο φάσεις. Αρχικά, ο $\mathcal{I}_{\mathcal{EL}}$ εφαρμόζεται εξαντλητικά στο $\mathcal{T}$, χρησιμοποιώντας ως κύριες προκείμενες μόνο τις RA-προτάσεις, και έπειτα, συλλέγονται όλες οι DL-Lite-προτάσεις από το $\mathcal{T}$, καθώς και όσες προέκυψαν από το προηγούμενο βήμα, ώστε να χρησιμοποιηθούν ως δευτερεύουσες στους κανόνες του $\mathcal{I}_{\mathcal{EL}}$ με κύριες προκείμενες προτάσεις τύπου 1. Όπως γίνεται κατανοητό στην περίπτωση ενός $\mathcal{T}$ που εκφράζεται σε DL-Lite, η πρώτη φάση παραλείπεται.

Παράδειγμα 4.1.4. Θεωρούμε το Παράδειγμα 4.1.2. Ο κανόνας ανάμεσα στις προτάσεις (4.1) και (4.2) που παράγει την (4.4) αντιστοιχίζεται σε κανόνα συνάρτησης.

Η πρόταση (4.6) μπορεί να παραχθεί από τον κανόνα συρρίκνωσης επί της (4.3) με δευτερεύουσες τις (4.2) και (4.4), ενώ η (4.7) παράγεται από τον κανόνα εκτύλιξης στο $\mathcal{Q}$ και την (4.6). $\diamondsuit$

Αλγόριθμος 2 Αλγόριθμος παραγωγής της επαναγραφής Rapid-EL$(\mathcal{Q}, \mathcal{T})$

Είσοδος: ένα ΣΕ, $\mathcal{Q}$ και ένα TBox, $\mathcal{T}$ που περιλαμβάνει προτάσεις.

1: $\mathcal{T}_{el} := \{\rho \mid \rho \in \mathcal{T}$ τ.ω. η ρ είναι της μορφής $A(u) \leftarrow R(u,v) \wedge B(v)\}$
2: $\mathcal{T}_{lt} := \mathcal{T} \setminus \{\mathcal{T}_{el} \cup \mathcal{T}_{eq}\}$
3: **repeat**
4: Επίλεξε μία πρόταση $\rho \in \mathcal{T}_{el}$
5: $\mathcal{P} :=$ REL$(\rho, \mathcal{T}_{lt})$ ▷ εκτύλιξη, συρρίκνωση, κανόνας συνάρτησης
6: $\mathcal{T}_{el} := \mathcal{T}_{el} \cup \{\rho \mid \rho \in \mathcal{P}$ τ.ω. η ρ είναι της μορφής $A(u) \leftarrow R(u,v) \wedge B(v)\}$
7: $\mathcal{T}_{lt} := \mathcal{T}_{lt} \cup (\mathcal{P} \setminus \mathcal{T}_{el})$
8: **until** λαμβάνοντας υπόψη τις μετονομασίες των μεταβλητών, δεν προστίθενται νέες προτάσεις στα $\mathcal{T}_{el}$, $\mathcal{T}_{lt}$
9: $\mathcal{R}_Q := \{\mathcal{Q}\}$
10: **repeat**
11: Επέλεξε ένα ερώτημα $\mathcal{Q}' \in \mathcal{R}_Q$
12: $\mathcal{R}_Q := \mathcal{R}_Q \cup$ RLite$(\mathcal{Q}', \mathcal{T}_{lt})$
13: **until** δεν προστίθεται κανένα νέο ερώτημα στο $\mathcal{R}_Q$ λαμβάνοντας υπόψη τις μετονομασίες των μεταβλητών
14: **return** $\mathcal{R}_Q \cup \mathcal{T}_{el}$

Από την αλγοριθμική διαδικασία που περιγράφεται παραπάνω, βλέπουμε πως στην περίπτωση της $\mathcal{ELHI}$, οι κανόνες της επίλυσης εφαρμόζονται αρχικά, στις RA-προτάσεις του σώματος ορολογίας, και έπειτα, εφαρμόζονται στα συζευκτικά ερωτήματα όπως και στην περίπτωση της DL-Lite. Πιο συγκεκριμένα, στη γραμμή 1 ορίζεται το υποσύνολο $\mathcal{T}_{el}$, του σώματος ορολογίας $\mathcal{T}$, που περιλαμβάνει όλες τις RA-προτάσεις, ενώ στη γραμμή 2, ορίζεται το υποσύνολο $\mathcal{T}_{lt}$, που περιλαμβάνει όλες τις DL-Lite προτάσεις του $\mathcal{T}$. Ο αλγόριθμος, στη γραμμή 4, επιλέγει μία RA-πρόταση από το $\mathcal{T}_{el}$ προκειμένου να εφαρμόσει του κανόνες του $\mathcal{I}_{\mathcal{EL}}$, μέσω της διαδικασίας REL$(\rho, \mathcal{T}_{lt})$. Στη γραμμή 6 το σύνολο $\mathcal{T}_{el}$ ενημερώνεται με τις νέες RA-πρότασεις που παράγονται από τη REL$(\rho, \mathcal{T}_{lt})$, και συγκεκριμένα, από την εφαρμογή του κανόνα της εκτύλιξης στην πρόταση ρ. Στη γραμμή 7 προστίθενται στο $\mathcal{T}_{lt}$ οι υπόλοιπες προτάσεις που παράγονται από τη διαδικασία REL$(\rho, \mathcal{T}_{lt})$, και συγκεκριμένα, οι (DL-Lite) προτάσεις που παράγονται από την εφαρμογή της συρρίκνωσης και του κανόνα-συνάρτησης στη ρ. Η ίδια διαδικασία επαναλαμβάνεται μέχρι να μην κατασκευάζονται νέες προτάσεις. Στη συνέχεια, ακολολουθεί η εφαρμογή των κανόνων του $\mathcal{I}_{lite}$ στο ΣΕ (γραμμή 12), λαμβάνοντας υπόψη το ενημερωμένο σύνολο $\mathcal{T}_{lt}$. Στην έξοδό του, ο αλγόριθμος επιστρέφει την ένωση των συζευκτικών ερωτημάτων που κατασκευάστηκαν, με τις RA-προτάσεις του $\mathcal{T}_{el}$.

4.2 Ορθότητα του $\mathcal{I}_{\mathcal{EL}}$

Όπως έχουμε ήδη αναφέρει, στο Κεφάλαιο 3, η ορθότητα του $\mathcal{I}_{lite}$ έγκειται στη δυνατότητα μετασχηματισμού μιας παραγωγής του $\mathcal{I}_{\mathsf{REQ}}$ σε μία παραγωγή του $\mathcal{I}_{lite}$. Εφόσον ο $\mathcal{I}_{lite}$ χρησιμοποιεί πάντοτε προτάσεις από το $\mathcal{T}$, δείξαμε αρχικά, πώς μία παραγωγή από το $\mathcal{I}_{\mathsf{REQ}}$ μπορεί να μετατραπεί σε SLD παραγωγή. Έπειτα, εφόσον οι $\mathcal{I}_{\mathsf{SLD}}, \mathcal{I}_{lite}$ παρουσιάζουν ιδιαίτερη ομοιότητα, δείξαμε πώς μία παραγωγή του $\mathcal{I}_{\mathsf{SLD}}$ μπορεί να μετασχηματιστεί σε παραγωγή του $\mathcal{I}_{lite}$. Με παρόμοιο τρόπο στην περίπτωση της $\mathcal{ELHI}$ δείχνουμε ότι οι παραγωγές από το $\mathcal{I}_{\mathsf{REQ}}$ μπορούν να αναχθούν σε παραγωγές με βάση το $\mathcal{I}_{\mathcal{EL}}$. Αναλύοντας τη δομή μίας παραγωγής που προκύπτει από το $\mathcal{I}_{\mathcal{EL}}$, καταφέρνουμε και πάλι να υιοθετήσουμε μία προσέγγιση δύο βημάτων.

Θεωρούμε ένα κανόνα συμπερασμού της μορφής $\Upsilon, \mathcal{C} \vdash \Upsilon'$, όπου $\mathcal{T} \vdash^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{C}$. Αν η παραγωγή της $\mathcal{C}$ δεν εμπλέκει κάποια RA-πρόταση, τότε, όπως και στην περίπτωση της DL-Lite, μπορούμε να δείξουμε ότι η Υ' μπορεί να προκύψει από την Υ με μία παραγωγή SLD χρησιμοποιώντας ως δευτερεύουσες προκείμενες μόνο DL-Lite προτάσεις. Για τους σκοπούς της απόδειξης μπορούμε να χρησιμοποιήσουμε και πάλι, την τεχνική εκτύλιξης του $\Upsilon, \mathcal{C} \vdash \Upsilon'$ σε $\Upsilon, \mathcal{C}_1 \vdash \Upsilon_2, \ldots, \Upsilon_{n-1}, \mathcal{C}_n \vdash \Upsilon'$, όπου για κάθε $1 \leq i \leq n$ ισχύει $\mathcal{C}_i \in \mathcal{T}$, και η $\mathcal{C}_i$ είναι DL-Lite-πρόταση. Σε διαφορετική περίπτωση, όπως εξηγήσαμε στην προηγούμενη ενότητα, ο λογισμός $\mathcal{I}_{\mathcal{EL}}$ παράγει προτάσεις χρησιμοποιώντας τους νέους κανόνες, που δεν επιτρέπουν κάποια RA-πρόταση στη θέση δευτερεύουσας προκείμενης. Προκειμένου να αντιστοιχιστούν σε παραγωγές που κατασκευάζονται από το $\mathcal{I}_{\mathcal{EL}}$, κάθε κανόνας θα πρέπει να ξεδιπλωθεί μέχρι ενός σημείου, δηλαδή, μέχρις ότου να προκύψει κάποια δευτερεύουσα προκείμενη $\mathcal{C}_k$, με $\mathcal{C}_k \notin \mathcal{T}$ που παράγεται από κάποια ακολουθία που ξεκινάει από μία RA-πρόταση. Εξετάζοντας προσεκτικά τον Ορισμό 4.1.3 παρατηρούμε εκτός από τις προτάσεις τύπου 1, ο $\mathcal{I}_{\mathcal{EL}}$ παράγει προτάσεις με μορφή που περιγράφεται παρακάτω, και άρα το ξεδίπλωμα πρέπει να σταματήσει σε οποιαδήποτε από τις ακόλουθες περιπτώσεις προτάσεων:

1. στην πρόταση $\mathcal{C}_k$ με τύπο 2.3 που είναι της μορφής $A(f(x)) \leftarrow B(x) \wedge [\mathbf{C}(x)]$ παράγεται από τον κανόνα συνάρτησης επί μίας RA-πρότασης, ή

2. στην $\mathcal{C}_k$ χωρίς συναρτήσεις με τύπο 3.3 που παράγεται από τους κανόνες συρρίκνωσης και εκτύλιξης ξεκινώντας από μία RA-πρόταση. Αυτό συμβαίνετι επείδή από μία RA-πρόταση της μορφής $A(x) \leftarrow R(x, y) \wedge \mathbf{B}(y)$ ο κανόνας συρρίκνωσης μπορεί να παράγει μία πρόταση της μορφής $A(x) \leftarrow C(x) \wedge [\mathbf{D}(x)]$, δηλαδή μία πρόταση με τύπο 3.3 χωρίς συναρτήσεις.

Από τα παραπάνω, συμπεραίνουμε ότι προκειμένου να αποδείξουμε την ορθότητα αρχικά, θα δείξουμε ότι οι παραγωγές από το $\mathcal{I}_{\mathsf{REQ}}$ μπορούν εν μέρη να ξεδιπλωθούν σε μία επεκταμένη μοφή από SLD παραγωγές οι οποίες μοιάζουν με αυτές που παράγονται από το $\mathcal{I}_{\mathcal{EL}}$. Οι εν λόγω SLD παραγωγές κατασκευάζονται εφαρμόζοντας κανόνες συμπερασμού με δευτερεύουσες προκείμενες DL-Lite προτάσεις από το $\mathcal{T}$, ή προτάσεις που παράγονται από τον κανόνα συνάρτησης (περίπτωση 1 από τα παραπάνω), ή προτάσεις χωρίς συνάρτηση τύπου 3.3 που έχουν παραχθεί στα προηγούμενα (περιπτωση 2 από τα παραπάνω).

Ορισμός 4.2.1. Έστω Σ ένα σύνολο από Horn προτάσεις. Μία *επεκταμένη* (*extended*) SLD παραγωγή από το Σ είναι μία ακολουθία από προτάσεις $\mathcal{C}_1, \ldots, \mathcal{C}_n$ τέτοια ώστε κάθε $\mathcal{C}_i$ μπορεί να είναι μία από τις παρακάτω:

- μία RA-πρόταση με τύπο 1 απο το Σ, ή
- το αποτέλεσμα ενός κανόνα που έχει κύρια πρόταση μία πρόταση με τύπο 1, ή 3.3, ή μία RA-πρόταση από το $\{\mathcal{C}_1, \ldots, \mathcal{C}_{i-1}\}$ και ως δευτερεύουσα μία πρόταση από το Σ, μία πρόταση με τύπο 3.3 χωρίς συναρτήσεις από το $\{\mathcal{C}_1, \ldots, \mathcal{C}_{i-1}\}$, ή μία πρόταση που παράγεται εφαρμόζοντας τον κανόνα συνάρτησης με δευτερεύουσες προεκίμενες από το Σ.

Ένα σύστημα που κατασεκυάζει τέτοιες παραγωγές δηλώνεται με $\mathcal{I}_{\mathsf{ESLD}}$. $\triangle$

Στα επόμενα δείχνουμε ότι κάθε παραγωγή από το $\mathcal{I}_{\mathsf{REQ}}$ μπορεί να μετασχηματιστεί σε μία επεκταμένη SLD παραγωγή.

Λήμμα 4.2.2. *Έστω $\mathcal{T}$ ένα σώμα ορολογίας σε $\mathcal{ELHI}$ και έστω Υ ένα ΣΕ (ή μία RA-πρόταση). Τότε κάθε πρόταση με τύπο 1 (αντίστοιχα με τύπο 3.3) που παράγεται από το $\mathcal{T}$ με το $\mathcal{I}_{\mathsf{REQ}}$ ξεκινώντας από την Υ, παράγεται επίσης, από το $\mathcal{T}$ με το $\mathcal{I}_{\mathsf{ESLD}}$ ξεκινώντας από την Υ.*

Απόδειξη. Αρχικά, έστω ότι η Υ είναι μία τυχαία RA-πρόταση που παράγει από το $\mathcal{T}$ με βάση το $\mathcal{I}_{\mathsf{REQ}}$, κάποια πρόταση με τύπο 3.3. Αποδεικνύουμε τον ισχυρισμό με επαγωγή. Έστω ότι σε κάποιο σημείο όλες οι προτάσεις με τύπο 3.3 που παράγονται από το $\mathcal{T}$ μέσω $\mathcal{I}_{\mathsf{REQ}}$ ξεκινώντας από μία RA-πρόταση παράγονται επίσης από το $\mathcal{T}$ μέσω $\mathcal{I}_{\mathsf{ESLD}}$ (IH1). Υποθέτουμε ότι μια νέα πρόταση $\mathcal{C}$ με τύπο 3.3 κατασκευάζεται σε κάποιο σημείο της παραγωγής που ξεκινάει με Υ. Δείχνουμε μόνο την περίπτωση που η Υ είναι της μορφής $A(x) \leftarrow R(x, y) \wedge \mathbf{D}(y)$, δηλαδή με τύπο 4.1. Η απόδειξη είναι παρόμοια στην περίπτωση που η Υ έχει τύπο 4.2.

Η πρόταση $\mathcal{C}$ είναι τύπου 3.3. Σύμφωνα με τον Πίνακα 2.5 τέτοιες προτάσεις μπορούν αν παραχθούν από κανόνες της μορφής $\mathcal{C}_1, \mathcal{D}_1 \vdash^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{C}$ όπου $\mathcal{C}_1$ και $\mathcal{D}_1$ έχουν μία από τις ακόλουθες μορφές:

1. Η πρόταση $\mathcal{C}_1$ έχει τύπο 4.1 και η πρόταση $\mathcal{D}_1$ έχει τύπο 2.1, η περίπτωση που η $\mathcal{C}_1$ έχει τύπο 4.2 και η $\mathcal{D}_1$ τύπο 2.2 είναι παρόμοια. Αφού ο $\mathcal{I}_{\mathsf{REQ}}$ δεν παράγει ποτέ προτάσεις με τύπο 4.1, τότε $\mathcal{C}_1 \in \mathcal{T}$. Αντίθετα, η $\mathcal{D}_1$ μπορεί να προκύψει από ένα κανόνα της μορφής $\mathcal{E}_2, \mathcal{D}_2 \vdash \mathcal{D}_1$, όπου η $\mathcal{E}_2$ είναι τύπου 3.1 και η $\mathcal{D}_2$ τύπου 2.1. Συνεπώς, όπως και στην απόδειξη του Λήμματος 3.2.1, ο κανόνας συμπερασμού $\mathcal{C}_1, \mathcal{D}_1 \vdash^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{C}$ μπορεί να ξεδιπλωθεί σε κανόνες $\mathcal{C}_1, \mathcal{E}_2 \vdash \mathcal{C}_2$ και $\mathcal{C}_2, \mathcal{D}_2 \vdash \mathcal{C}$. Οι προτάσεις με τύπο 3.1 δεν παράγονται ποτέ από το $\mathcal{I}_{\mathsf{REQ}}$, και έτσι $\mathcal{E}_2 \in \mathcal{T}$, ενώ η $\mathcal{D}_2$ (με τύπο 2.1) έχει μικρότερο βάθος παραγωγής. Επαγωγικά δείχνουμε ότι μπορούμε να ξεδιπλώσουμε τους κανόνες μέχρι να φτάσουμε σε παραγωγή της μορφής $\mathcal{C}_1, \ldots, \mathcal{C}_n, \mathcal{C}$ όπου $\mathcal{C}_i, \mathcal{E}_i \vdash \mathcal{C}_{i+1}$, όλες οι $\mathcal{C}_i$ έχουν τύπο 4.1, $\mathcal{C}_1 \in \mathcal{T}$, όλες $\mathcal{E}_i$ έχουν τύπο 3.1 (ή 3.2) και είναι στο $\mathcal{T}$, ενώ η $\mathcal{C}$ προκύπτει από κάποιον $\mathcal{C}_n, \mathcal{D}_n \vdash \mathcal{C}$, όπου $\mathcal{D}_n$ έχει τύπο 2.1 (ή 2.2) και είναι στο $\mathcal{T}$. Συνεπώς, η $\mathcal{C}$ παράγεται από το $\mathcal{T}$ μέσω $\mathcal{I}_{\mathsf{ESLD}}$.

2. Η πρόταση $\mathcal{C}_1$ έχει τύπο 3.3 και η $\mathcal{D}_1$ είναι κάποια πρόταση. Αφού η $\mathcal{C}$ παράγεται από μια ακολουθία που ξεκινάει από την Υ, τότε το ίδιο συμβαίνει και με την $\mathcal{C}_1$. Έτσι, από την επαγωγική υπόθεση IH1, η $\mathcal{C}_1$ παράγεται από το $\mathcal{T}$ με $\mathcal{I}_{\mathsf{ESLD}}$. Εστιάζουμε τώρα την προσοχή μας στο $\mathcal{D}_1$. Από μία δεύτερη επαγωγή μπορούμε να δείξουμε ότι αν η $\mathcal{D}_1$ παράγεται από εφαρμογή κανόνων σε άλλες προτάσεις, τότε ο $\mathcal{C}_1, \mathcal{D}_1 \vdash^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{C}$ μπορεί είτε να ξεδιπλωθεί χρησιμοποιώντας αυτές τις προτάσεις που έχουν χαμηλότερο βάθος στην παραγωγή είτε η $\mathcal{D}_1$ μπορεί να παραχθεί με το $\mathcal{I}_{\mathsf{ESLD}}$. Πιο συγκεκριμένα έχουμε τον παρακάτω ισχυρισμό:

> *Ισχυρισμός 3*: Για κάθε $\mathcal{C}_1, \mathcal{C}$, και $\mathcal{D}_1$ τέτοιες ώστε $\mathcal{C}_1, \mathcal{D}_1 \vdash \mathcal{C}$ και $\vdash_j^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{D}_1$ έχουμε ότι αν η $\mathcal{D}_1$ έχει τύπο 3.3., τότε δεν περιλαμβάνει συνάρτηση και επιπλέον είτε η $\mathcal{D}_1$ παράγεται από το $\mathcal{I}_{\mathsf{ESLD}}$, είτε υπάρχουν προτάσεις $\mathcal{F}$ και $\mathcal{G}$ τέτοιες ώστε $\vdash_{j-1}^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{F}$, $\vdash_{j-1}^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{G}$, και $\mathcal{C}_1, \mathcal{F} \vdash \mathcal{C}', \mathcal{C}', \mathcal{G} \vdash \mathcal{C}$

Απόδειξη του Ισχυρισμού 3: Μελετάμε διαφορετικές περιπτώσεις ανάλογα με τη μορφή της $\mathcal{D}_1$:

(α′) $\mathcal{D}_1$ με τύπο 2.3. Τέτοιες προτάσεις μπορούν να παραχθούν από κανόνες της μορφής $\mathcal{F}, \mathcal{G} \vdash^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{D}_1$, όπου $\mathcal{F}$ και $\mathcal{G}$ μπορούν να έχουν ένα από τους παρακάτω τύπους:

i. $\mathcal{F}$ με τύπο 4.1 (4.2) και $\mathcal{G}$ με τύπο 2.2 (2.1). Δείχνουμε μόνο την περίπτωση 4.1+2.2.
Αρχικά, υποθέτουμε ότι η $\mathcal{F}$ περιλαμβάνει ένα άτομο $[C(y)]$, είναι δηλαδή της μορφής $A(x) \leftarrow R(x, y) \wedge \mathbf{C}(y)$. Τότε, ο κανόνας $\mathcal{F}, \mathcal{G} \vdash^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{D}_1$ αντιστοιχίζεται σε ένα κανόνα συνάρτησης. Επιπλέον, αφού η $\mathcal{F}$ έχει τύπο 4.1 έχουμε ότι $\mathcal{F} \in \mathcal{T}$. Συνεπώς, αν επιπρόσθετα $\mathcal{G} \in \mathcal{T}$, τότε η $\mathcal{D}_1$ μπορεί να παραχθεί με $\mathcal{I}_{\mathsf{ESLD}}$ όπως απαιτείται. Αντίθετα, η $\mathcal{G}$ μπορεί να παραχθεί από ένα κανόνα που εφαρμόζεται σε μία πρόταση $\mathcal{J}_1$ με τύπο 3.1 και μία άλλη $\mathcal{G}_1$ με τύπο 2.2. Τότε, ο κανόνας $\mathcal{F}, \mathcal{G} \vdash \mathcal{D}_1$ μπορεί να ξεδιπλωθεί στους $\mathcal{F}, \mathcal{J}_1 \vdash \mathcal{F}_1$ και $\mathcal{F}_1, \mathcal{G}_1 \vdash \mathcal{D}_1$. Επιπρόσθετα, έχουμε ότι $\mathcal{J}_1 \in \mathcal{T}$ αφού τέτοιες προτάσεις δεν περάγονται ποτέ από το $\mathcal{I}_{\mathsf{REQ}}$ και έτσι η $\mathcal{F}_1$ μπορει να παραχθεί με $\mathcal{I}_{\mathsf{SLD}}$ από την $\mathcal{F}$. Αυτό μπορεί να συμβαίνει εξαντλητικά μέχρις ότου προκύψει μία παραγωγή $\mathcal{F}_0, \mathcal{F}_1, \ldots, \mathcal{F}_n, \mathcal{D}_1$ με $0 \leq i < n$ όπου έχουμε $\mathcal{F}_i, \mathcal{J}_{i+1} \vdash \mathcal{F}_{i+1}$, $\mathcal{F}_0 = \mathcal{F}$, όλες οι $\mathcal{J}_{i+1}$ έχουν τύπο 3.1, και $\mathcal{F}_n, \mathcal{G}_n \vdash \mathcal{C}_i$ για κάποια $\mathcal{G}_n \in \mathcal{T}$. Εφόσον όλες οι $\mathcal{J}_i$ έχουν τύπο 3.1 και $\mathcal{F}_0 \in \mathcal{T}$, τότε η $\mathcal{F}_n$ παράγεται με $\mathcal{I}_{\mathsf{SLD}}$. Επιπλέον, ο τελευταίος κανόνας σε κάποια ακολουθία ανταποκρίνεται στον κανόνα συνάρτησης και εφόσον η $\mathcal{F}_n$ παράγεται με το $\mathcal{I}_{\mathsf{SLD}}$ και $\mathcal{G}_n \in \mathcal{T}$, τότε καταλήγουμε ότι η $\mathcal{D}_1$ μπορεί να παραχθεί από το $\mathcal{T}$ με το $\mathcal{I}_{\mathsf{ESLD}}$.
Δεύτερον, υποθέτουμε ότι η $\mathcal{F}$ δεν περιλαμβάνει το άτομο $[\mathbf{C}(y)]$, είναι δηλαδή της μορφής $A(x) \leftarrow R(x, y)$. Τότε, ο κανόνας $\mathcal{F}, \mathcal{G} \vdash \mathcal{D}_1$ μπορεί να χρησιμοποιηθεί για να ξεδιπλώσουμε τον κανόνα $\mathcal{C}_1, \mathcal{D}_1 \vdash \mathcal{C}$

στους $\mathcal{C}_1, \mathcal{F} \vdash \mathcal{C}'$ και $\mathcal{C}', \mathcal{G} \vdash \mathcal{C}$, όπου οι $\mathcal{G}$ και $\mathcal{F}$ παράγονται σε χαμηλότερο βάθος.

ii. η $\mathcal{F}$ έχει τύπο 3.3, η $\mathcal{G}$ τύπο 2.3 και σύμφωνα με τον Πίνακα 2.5 η $\mathcal{F}$ δεν περιλαμβάνει συναρτήσεις. Σε αυτή την περίπτωση ο κανόνας $\mathcal{C}_1, \mathcal{D}_1 \vdash \mathcal{C}$ μπορεί να ξεδιπλωθεί στους $\mathcal{C}_1, \mathcal{F} \vdash \mathcal{C}'$ $\mathcal{C}', \mathcal{G} \vdash \mathcal{D}_1$ και οι $\mathcal{F}$, $\mathcal{G}$ προφανώς έχουν μικρότερα βάθη παραγωγής και σύμφωνα με τον Πίνακα 2.5 η πρόταση $\mathcal{F}$ δεν περιλαμβάνει συναρτήσεις. Έτσι, όλες οι συνθήκες του *Ισχυρισμού 3* ικανοποιούνται.

iii. οι $\mathcal{F}$ και $\mathcal{G}$ έχουν τύπο 2.3. Τότε, ο κανόνας $\mathcal{C}_1, \mathcal{D}_1 \vdash \mathcal{C}$ μπορεί να ξεδιπλωθεί σε κανόνες της μορφής $\mathcal{C}_1, \mathcal{F} \vdash \mathcal{C}'$, $\mathcal{C}', \mathcal{G} \vdash \mathcal{C}$ όπου $\mathcal{F}$ και $\mathcal{G}$ παράγονται σε χαμηλότερο βάθος.

(β′) η $\mathcal{D}_1$ έχει τύπο 3.3. Από την επαγωγική υπόθεση του *Ισχυρισμού 3* προκύπτει ότι η $\mathcal{D}_1$ δεν περιλαμβάνει συναρτήσεις, και άρα είναι της μορφής $A(x) \leftarrow B(x) \wedge [\mathbf{C}(x)]$. Έχουμε τώρα τις παρακάτω δύο περιπτώσεις:

Αρχικά υποθέτουμε ότι $\mathcal{D}_1$ είναι $A(x) \leftarrow B(x)$ (δεν περιλαμβάνει το άτομο $C(x)$). Τότε, για την $\mathcal{F}$ και $\mathcal{G}$ έχουμε τις παρακάτω περιπτώσεις:

i. η $\mathcal{F}$ είναι της μορφής $A(x) \leftarrow R(x, y)$ (τύπος 4.1 χωρίς το $[\mathbf{C}(y)]$) και η $\mathcal{G}$ τηε μορφής $R(x, f(x)) \leftarrow B(x)$ (τύπος 2.1) ή $\mathcal{F}$ της μορφής $A(x) \leftarrow R(y, x)$ (τύπος 4.2 και πάλι χωρίς το $[\mathbf{C}(y)]$) και $\mathcal{G}$ είναι της μορφής $R(f(x), x) \leftarrow B(x)$ (τύπος 2.2). Σε αυτή την περίπτωση μπορούμε να ξεδιπλώσουμε τον κανόνα $\mathcal{C}_1, \mathcal{D}_1 \vdash \mathcal{C}$ σε $\mathcal{C}_1, \mathcal{F} \vdash \mathcal{C}'$, $\mathcal{C}', \mathcal{G} \vdash \mathcal{C}$ όπου $\mathcal{F}$ και $\mathcal{G}$ παράγονται σε μικρότερο βάθος.

ii. η $\mathcal{F}$ έχει τη μορφή $A(x) \leftarrow D(f(x)) \wedge B(x)$ (τύπος 3.3) και η $\mathcal{G}$ τη μορφή $D(f(x)) \leftarrow B(x)$ (τύπος 2.3). Εφόσον το σώμα ορολογίας $\mathcal{T}$ δεν περιλαμβάνει ποτέ προτάσεις της μορφής $A(x) \leftarrow D(f(x)) \wedge B(x)$, τότε η $\mathcal{F}$ πρέπει να έχει παραχθεί εφαρμόζοντας το $\mathcal{I}_{\mathsf{REQ}}$ στο $\mathcal{T}$. Η $\mathcal{F}$ παράγεται από ένα κανόνα από τους παραπάνω, 3.3+2.3, όπου η κύρια πρόταση πρέπει να περιλαμβάνει έναν συναρτησιακό όρο στο σώμα της. Συνεπώς, και πάλι η πρόταση με τύπο 3.3 δεν μπορεί να ανήκει στο $\mathcal{T}$. Μπορούμε να καταλήξουμε ότι αυτή είναι η περίπτωση που η παραγωγή που κατασκευάζει την $\mathcal{D}_1$ πρέπει να ξεκινά με μία RA-πρόταση. Τότε όμως, από την επαγωγική υπόθεση IH1, έχουμε ότι η $\mathcal{D}_1$ έχει προκύψει από το $\mathcal{T}$ με $\mathcal{I}_{\mathsf{ESLD}}$.

Έπειτα, θεωρούμε ότι η $\mathcal{D}_1$ είναι της μορφής $A(x) \leftarrow B(x) \wedge \mathbf{C}(x)$ (δηλαδή περιλαμβάνει ένα άτομο $\mathbf{C}(x)$). Πάλι όπως και στην προηγούμενη περίπτωση, μπορούμε να συμπαιράνουμε από τη μορφή των κανόνων ότι η $\mathcal{D}_1$ προκύπτει από μία παραγωγή που ξεκινά από μία RA-πρόταση. Από την επαγωγική υπόθεση IH1 η $\mathcal{D}_1$ παράγεται από το $\mathcal{T}$ με τον $\mathcal{I}_{\mathsf{ESLD}}$.

Εδώ ολοκληρώνεται η περίπτωση όπου η Υ είναι μία RA-πρόταση.

Έπειτα, αποδεικνύουμε την περίπτωση που η Υ είναι τύπου 1 $\mathcal{Q}$ και παράγει μία άλλη πρόταση $\mathcal{Q}'$ τύπου 1—δηλαδή για $\mathcal{Q}, \mathcal{Q}'$ έχουμε ότι $\mathcal{T}, \mathcal{Q} \vdash_i^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{Q}'$.

Έστω $\mathcal{P}_{lt}$ το σύνολο με όλες τις DL-Lite-προτάσεις που παράγονται από το $\mathcal{T}$ με βάση το $\mathcal{I}_{\mathsf{ESLD}}$. Δείχνουμε με επαγωγή ότι $\mathcal{P}_{lt}, \mathcal{Q} \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}'$.

Βασική περίπτωση (i=0): Τότε $\mathcal{Q}' = \mathcal{Q}$ και ισχύει $\mathcal{P}_{lt}, \mathcal{Q} \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}$.

Επαγωγικό βήμα: Υποθέτουμε ότι για κάθε $\mathcal{Q}_i$ τέτοιο ώστε $\mathcal{T}, \mathcal{Q} \vdash_i^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{Q}_i$ έχουμε ότι $\mathcal{P}_{lt}, \mathcal{Q} \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}_i$ (επαγωγική υπόθεση). Υποθέτουμε τώρα ότι σε επόμενο βήμα παράγεται μία πρόταση $\mathcal{Q}_{i+1}$ με τύπο 1, δηλαδή , $\mathcal{Q} \vdash_{i+1}^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{Q}_{i+1}$. Από τον ορισμό του $\mathcal{I}_{\mathsf{REQ}}$ το $\mathcal{Q}_{i+1}$ παράγεται από ένα κανόνα της μορφής $\mathcal{Q}_i, \mathcal{C} \vdash^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{Q}_{i+1}$, που έχει κύρια πρόταση μία πρόταση τύπου 1 τέτοια ώστε $\mathcal{T}, \mathcal{Q} \vdash_i^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{Q}_i$ και σα δευτερεύουσα μία πρόταση $\mathcal{C}$ τέτοια ώστε $\mathcal{T} \vdash^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{C}$. Προφανώς έχουμε $\mathcal{Q}_i, \mathcal{C} \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}_{i+1}$. Αν $\mathcal{C} \in \mathcal{T}$ τότε από την επαγωγική υπόθεση και αφού το $\mathcal{P}_{lt}$ αρχικοποιείται με το $\mathcal{T}$ προκύπτει άμεσα $\mathcal{P}_{lt}, \mathcal{Q} \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}_{i+1}$. Σε διαφορετική περίπτωση έχουμε $\mathcal{T} \vdash_j^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{C}$ με $j > 0$ και χρειάζεται να δείξουμε είτε ότι $\mathcal{C} \in \mathcal{P}_{lt}$ ή ότι το $\mathcal{Q}_{i+1}$ μπορεί επίσης να προκύψει από το $\mathcal{Q}_i$ με $\mathcal{I}_{\mathsf{SLD}}$ χρησιμοποιώντας μόνο προτάσεις του $\mathcal{P}_{lt}$. Αυτό προκύπτει από τον ακόλουθο ισχυρισμό:

> *Ισχυρισμός 4*: Για κάθε $\mathcal{Q}_{i+1}$ και $\mathcal{C}$ τέτοια ώστε $\mathcal{Q}_i, \mathcal{C} \vdash \mathcal{Q}_{i+1}$, $\mathcal{T} \vdash_j^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{C}$, $j > 0$, είτε $\mathcal{C} \in \mathcal{P}_{lt}$ ή $\mathcal{C} \notin \mathcal{P}_{lt}$ και ισχύουν οι ακόλουθες συνθήκες:
>
> - Υπάρχουν προτάσεις $\mathcal{C}_1$ και $\mathcal{C}_2$ τέτοιες ώστε $\mathcal{T} \vdash_{j-1}^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{C}_1, \mathcal{T} \vdash_{j-1}^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{C}_2$, και $\mathcal{Q}_i, \mathcal{C}_1 \vdash \mathcal{Q}'$, $\mathcal{Q}', \mathcal{C}_2 \vdash \mathcal{Q}_{i+1}$.
> - Αν η $\mathcal{C}$ έχει τύπο 3.3 τότε δεν περιλαμβάνει συναρτήσεις.

Αποδεικνύουμε τον Ισχυρισμό 4 από την επίλυση περιπτώσεων στους τύπους των προτάσεων που μπορούν να παραχθούν από το $\mathcal{T}$ με το $\mathcal{I}_{\mathsf{REQ}}$, δηλαδή στους πιθανούς τύπους της πρότασης $\mathcal{C}$:

1. Η $\mathcal{C}$ έχει τύπο 2.1 ή τύπο 2.2. Τότε, από τον Πίνακα 2.5 τέτοιες προτάσεις μπορούν να προκύψουν από κανόνες που εμπλέκουν μόνο DL-Lite-προτάσεις (δηλαδή κανόνες της μορφής 3.1+2.1=2.1 και 3.1+2.2=2.2). Έτσι, ο ισχυριμός προκύπτει από την απόδειξη του ισχυρισμού 1 του Λήμματος 1 3.2.1.

2. Η $\mathcal{C}$ είναι τύπου 2.3 ή τύπου 3.3. Αυτές οι περιπτώσεις είναι παρόμοιες με τις 2a και 2b στην απόδειξη του *Ισχυρισμού 3*, και έτσι παραλείπουμε τις λεπτομέρειες.

Ο προηγούμενος ισχυρισμός υπονοεί ότι για μία πρόταση $\mathcal{C} \notin \mathcal{P}_{lt}$ τέτοια ώστε $\mathcal{T} \vdash_j^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{C}$, και $j > 0$, ο κανόνας $\mathcal{Q}_i, \mathcal{C} \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}_{i+1}$ μπορεί να μετασχηματιστεί σε μία ακολουθία από κανόνες της μορφής $\mathcal{Q}_i, \mathcal{C}_1 \vdash \mathcal{Q}'_1$, $\mathcal{Q}'_1, \mathcal{C}_2 \vdash \mathcal{Q}'_2, \ldots, \mathcal{Q}'_{n-1}, \mathcal{C}_n \vdash \mathcal{Q}_{i+1}$, τέτοια ώστε για όλα τα $1 \leq i \leq n$ να έχουμε $\mathcal{C}_i \in \mathcal{P}_{lt}$ και επομένως $\mathcal{P}_{lt}, \mathcal{Q}_i \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}_{i+1}$ και λαμβάνοντας υπόψη την επαγωγική υπόθεση ($\mathcal{P}_{lt}, \mathcal{Q} \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}_i$) προκύπτει το ζητούμενο $\mathcal{P}_{lt}, \mathcal{Q} \vdash^{\mathcal{I}_{\mathsf{SLD}}} \mathcal{Q}_{i+1}$.

□

Επιπλέον, προκύπτει από τη δομή μιας επεκτεμένης SLD παραγωγής ότι οι δευτερεύουσες προτάσεις μπορούν να περιλαμβάνουν συναρτησιακά σύμβολα μόνο στην κεφαλή.

Πρόταση 4.2.3. *Έστω $\mathcal{T}$ ένα σώμα ορολογίας και $\mathcal{Q}$ ένα ερώτημα. Κάθε επεκταμένη SLD παραγωγή από το $\mathcal{T} \cup \{\mathcal{Q}\}$ χρησιμοποιεί ως δευτερεύουσες προκείμενες προτάσεις που δεν περιλαμβάνουν συναρτησιακά σύμβολα παραμόνο στην κεφαλή.*

Απόδειξη. Από τον Ορισμό 4.2.1 μία επεκταμένη SLD παραγωγή χρησιμοποιεί ως δευτερεύουσες είτε προτάσεις από το $\mathcal{T}$, που περιέχουν συναρτησιακά σύμβολα μόνο στην κεφαλή, είτε προτάσεις που παράγονται από τον κανόνα της συνάρτησης, οι οποίες από τον Ορισμό 4.1.3 περιλαμβάνουν σύμβολα συνάρτησης μόνο στην κεφαλή, ή προτάσεις με τύπο 3.3 χωρίς συναρτήσεις. □

Συνεπώς, το Λήμμα 3.2.3 βρίσκει εφαρμογή και κάθε επεκταμένη SLD παραγωγή μιας πρότασης χωρίς συναρτήσεις με τύπο 1 ή τύπο 3.3. μπορεί να μετατραπεί σε άλλες που είναι συμπαγείς ως προς τη συνάρτηση. Αποδεικνύουμε σε αυτό το σημείο την ορθότητα του $\mathcal{I}_{\mathcal{EL}}$.

Θεώρημα 4.2.4. *Έστω ένα σώμα ορολογίας $\mathcal{T}$ σε $\mathcal{ELHI}$ και ένα ΣΕ $\mathcal{Q}$. Κάθε παραγωγή του $\mathcal{I}_{\mathcal{EL}}$ από το $\mathcal{T} \cup \{\mathcal{Q}\}$ τερματίζει. Επιπλέον, το* Rapid-EL$(\mathcal{Q}, \mathcal{T})$ *είναι μια datalog επαναγραφή του $\mathcal{Q}$ με βάση το $\mathcal{T}$.*

Απόδειξη. Χρησιμοποιώντας τους ίδιους ισχυρισμούς όπως και στην απόδειξη του Θεωρήματος 3.2.5 αποδεικνύουμε ότι η διαδικασία επαναγραφής τερματίζει. Η ορθότητα προκύπτει άμεσα από την ορθότητα του συστήματος κανόνων $\mathcal{I}_{\mathcal{EL}}$, ενώ πρέπει να αποδείξουμε την πληρότητα του λογισμού. Έστω $\mathcal{R} =$ Rapid-EL$(\mathcal{Q}, \mathcal{T})$. Θεωρούμε την έξοδο του Requiem $\mathcal{R}_r \in$ RQR$(\mathcal{Q}, \mathcal{T})$. Το $\mathcal{R}_r$ μπορεί να διαμεριστεί σε $\mathcal{R}_D$ και $\mathcal{R}_Q$ ικανοποιώντας τις συνθήκες του Ορισμού 2.4.1.

Απο το Λήμμα 4.2.2 κάθε παραγωγή μίας πρότασης $\mathcal{Q}'$ τύπου 1 από το $\mathcal{T} \cup \{\mathcal{Q}\}$ με το $\mathcal{I}_{\mathsf{REQ}}$ μπορεί να μετασχηματιστεί σε μία παραγωγή SLDP. Επίσης, από την Πρόταση 4.2.3 η παραγωγή της $\mathcal{Q}'$ μπορεί να μετασχηματιστεί σε μία άλλη συμπαγή ως προς τις συναρτήσεις. Έτσι, προκύπτει με τρόπο παρόμοιο όπως και στην απόδειξη του Θεωρήματος 3.2.5 ότι κάθε $\mathcal{Q}'$ που παράγεται από το $\mathcal{T} \cup \{\mathcal{Q}\}$ με το $\mathcal{I}_{\mathsf{REQ}}$, ανήκει επίσης στο $\mathcal{R}$.

Τώρα θεωρούμε τους κανόνες datalog που παράγονται από το $\mathcal{I}_{\mathsf{REQ}}$. Έχουν μία από τις παρακάτω μορφές (έχουμε παραλείψει τις ανάλογες περιπτώσεις με τους αντίστροφους ρόλους):

$$A(x) \leftarrow R(x,y) \wedge C(y) \tag{4.8}$$

$$A(x) \leftarrow R(x,y) \tag{4.9}$$

$$R(x,y) \leftarrow S(x,y) \tag{4.10}$$

$$A(x) \leftarrow B(x) \wedge [C(x)] \tag{4.11}$$

Οι προτάσεις της μορφής (4.8)–(4.10) δεν παράγονται ποτέ από το $\mathcal{I}_{\mathsf{REQ}}$. Έτσι, προκύπτει ότι όλοι οι κανόνες ανήκουν επίσης στο $\mathcal{R}$. Οι προτάσεις της μορφής (4.11) είτε εμφανίζονται στο $\mathcal{T}$ είτε παράγονται από ακολουθίες που ξεκινούν μία RA-πρόταση όπως φαίνεται στο Λήμμα 4.2.2. Στην τελευταία περίπτωση, προκύπτει από το Λήμμα 4.2.2 και την Πρόταση 4.2.3 ότι οι προτάσεις αυτές μπορούν να παραχθούν από τους κανόνες εκτύλιξης και συρρίκνωσης, και άρα, ανήκουν επίσης στο $\mathcal{R}$.

Τελικά, το $\mathcal{R}_D$ παράγεται από το βήμα εκτύλιξης του Requiem χρησιμοποιώντας τους κανόνες datalog που έχουν παραχθεί και το $\mathcal{R}_Q$ παράγεται εκτυλίσσοντας τα ερωτήματα με τις προτάσεις που έχουν παραχθεί. Όπως δείξαμε προηγουμένως, το $\mathcal{R}_D$ είναι ισοδύναμο με το datalog κομμάτι του $\mathcal{R}$. Επίσης, όπως και στο Θεώρημα 3.2.5 όλοι οι κανόνες εκτύλιξης πάνω στα ερωτήματα μπορούν να μετασχηματιστούν σε κανόνες με δευτερεύουσες προκείμενες από το $\mathcal{T}$, ή προτάσεις με τύπο 3.3 χωρίς συναρτήσεις από το $\mathcal{R}$. Επομένως $\mathcal{R}_Q \subseteq \mathcal{R}$.

□

Κεφάλαιο 5

Επαναγραφή στη Horn-$\mathcal{SHIQ}$

Σε αυτή την ενότητα περιγράφουμε ένα βελτιστοποιημένο αλγόριθμο επαναγραφής για τη γλώσσα Horn-$\mathcal{SHIQ}$. Αρχικά, παρουσιάζουμε έναν αλγόριθμο βασικής υπέρθεσης, ο οποίος εφαρμόζει κατάλληλη συνάρτηση επιλογής με σκοπό να αποτρέπει τη διάδοση των συναρτησιακών συμβόλων στις παραγόμενες προτάσεις. Στη συνέχεια, βελτιστοποιούμε τον αλγόριθμο επεκτείνοντας το σύστημα κανόνων $\mathcal{I}_{\mathcal{EL}}$ για την $\mathcal{ELHI}$. Στην τελευταία ενότητα του κεφαλαίου, αποδεικνύουμε την ορθότητα του βελτιστοποιημένου αλγορίθμου.

5.1 Κανόνες βασικής υπέρθεσης του $\mathcal{I}_{\mathrm{H}\mathcal{S}}$

Όπως έχουμε ήδη αναφέρει (Ενότητα 2.4.1), οι κανόνες της βασικής υπέρθεσης στην περίπτωση της Horn-$\mathcal{SHIQ}$, ενδέχεται να παράγουν προτάσεις με συναρτησιακά σύμβολα που πρέπει να ληφθούν υπόψη στο σύνολο της επαναγραφής. Κάτι τέτοιο δεν ισχύει για τις προτάσεις προτάσεις (2.13) και (2.14) του Παραδείγματος 2.4.4 καθώς δεν επιλύονται με βασικές προτάσεις του σώματος ισχυρισμών και άρα, δε συμπεριλαμβάνονται στην έξοδο του αλγορίθμου επαναγραφής. Επιπλέον, οι συγκεκριμένες προτάσεις δε συνεισφέρουν στην κατασκευή άλλων προτάσεων που είναι μέλη της επαναγραφής. Επομένως, όπως γίνεται κατανοητό, οι κανόνες συμπερασμού που οδήγησαν στην παραγωγή των προτάσεων (2.13) και (2.14) θα ήταν προτιμότερο να μην είχαν εφαρμοστεί. Πράγματι, οι άσκοπες εφαρμογές των κανόνων επίλυσης αποτελούν ένα από τα βασικότερα προβλήματα των αλγορίθμων επαναγραφής με επίλυση [122], ιδιαίτερα στις περιπτώσεις μεγάλων οντολογιών.

Σαν αφετηρία για το σχεδιασμό ενός αλγορίθμου βασικής υπέρθεσης χρησιμοποιήσαμε τον αλγόριθμο που υλοποιήθηκε στο σύστημα KAON2 [70], και περιγράψαμε στην Ενότητα 2.2. Σημειώνουμε ότι ο αλγόριθμος του KAON2 αποφαίνεται τη συνεπαγωγή βασικών προτάσεων (χωρίς μεταβλητές) από μία δεδομένη βάση γνώσης. Στην παρούσα εργασία σχεδιάζουμε έναν αλγόριθμο ο οποίος κατασκευάζει την επαναγραφή μιας οντολογίας και ενός συζευκτικού ερωτήματος. Παρόμοια με την προσέγγισή μας για τις γλώσσες DL-Lite και $\mathcal{ELHI}$, επιχειρούμε να σχεδιάσουμε έναν αλγόριθμο ο οποίος να εφαρμόζει μόνο κανόνες συμπερασμού που παράγουν απαραίτητες, για την επαναγραφή, προτάσεις.

Σαν ένα πρώτο βήμα προς αυτή την κατεύθυνση ορίζουμε μία συνάρτηση επιλογής διαφορετική από αυτή που υιοθετείται στο [70]. Ο αλγόριθμος του KAON2 δίνει προτεραιότητα στην επίλυση των ατόμων που είναι μέγιστα στην πρόταση και όχι στην απομάκρυνση των ατόμων με συναρτήσεις. Αν ανατρέξουμε στο Παράδειγμα 2.4.4, παρατηρούμε ότι ο κανόνας της υπέρθεσης εφαρμόζεται στην πρόταση (2.14), χωρίς να λαμβάνεται υπόψη ότι η πρόταση (2.13) περιέχει το άτομο $K(h(x))$, που πρέπει να απομακρυνθεί σε επόμενα βήματα ώστε να προκύψει πρόταση χωρίς συναρτήσεις στο σώμα. Επομένως, ορίζουμε μία νέα συνάρτηση επιλογής που δίνει προτεραιότητα στα άτομα με συναρτήσεις. Επιπλέον, προκειμένου η διάδοση των συναρτησιακών συμβόλων στις προτάσεις που παράγονται, να είναι ελεγχόμενη, ο κανόνας της υπερεπίλυσης αντικαθίσταται από τον κανόνα της δυαδικής επίλυσης. Με αυτόν τον τρόπο η συμπερασματική πρόταση περιλαμβάνει ένα μοναδικό συναρτησιακό σύμβολο που θα πυροδοτήσει την εφαρμογή του επόμενου κανόνα επίλυσης.

Ορισμός 5.1.1. Με $\mathcal{I}_{\mathrm{HS}}$ συμβολίζουμε το λογισμό βασικής υπέρθεσης για τη Horn-$\mathcal{SHIQ}$, που περιλαμβάνει τους κανόνες θετικής υπέρθεσης και επίλυσης, με παραμέτρους μία διάταξη $\succ$ ολική στους βασικούς όρους, και μία συνάρτηση επιλογής η οποία

- επιλέγει το άτομο στο σώμα της πρότασης το οποίο περιλαμβάνει συνάρτηση και έχει το μεγαλύτερο βάθος
- εάν κανένα άτομο στο σώμα δεν περιλαμβάνει συνάρτηση, επιλέγει κάποιο ρόλο που βρίσκεται στο σώμα της πρότασης, εαν υπάρχει, διαφορετικά δεν επιλέγει τίποτα.

△

Παράδειγμα 5.1.2. Θεωρούμε το σώμα ορολογίας $\mathcal{T}$ του Παραδείγματος 2.4.4. Όταν ο $\mathcal{I}_{\mathrm{HS}}$ εφαρμόζεται στο $\mathcal{T}$ πραγματοποιεί τα παρακάτω βήματα: πρώτα μία συνάρτηση επιλογής επιλέγει οποιοδήποτε από τα δύο άτομα ρόλου στο σώμα της (2.8), έστω το $R(x, y)$, και πραγματοποιεί τον κανόνα επίλυσης:

$$(2.8), (2.9) \vdash f(x) \approx z \leftarrow K(x) \wedge R(x, z) \wedge B(f(x)) \wedge B(z) \tag{5.1}$$

Έπειτα, επιλέγεται το άτομο $B(f(x))$ που έχει το μεγαλύτερο βάθος, στην πρόταση (5.1):

$$(5.1), (2.10) \vdash f(x) \approx z \leftarrow K(x) \wedge R(x, z) \wedge B(z) \tag{5.2}$$

Εφόσον η πρόταση (5.2) δεν περιλαμβάνει συναρτήσεις στο σώμα, επιλέγεται το άτομο $R(x, y)$ και έτσι η πρόταση (5.2) επιλύεται με την (2.11) προκειμένου να παραχθεί η (2.13) του Παραδείγματος 2.4.4. Όμως, αντίθετα από ότι συμβαίνει στο Παράδειγμα 2.4.4, ο αλγόριθμος επιλέγει το $K(h(x))$ της (2.13) και έτσι, δεν εφαρμόζεται ο κανόνας της υπέρθεσης με την (2.9) ◇

Από τη θεωρία της υπέρθεσης είναι γνωστό ότι η συνάρτηση επιλογής με την οποία παραμετροποιείται η μέθοδος, μπορεί τυχαία να επιλέγει οποιοδήποτε από τα αρνητικά

άτομα, δηλαδή τα άτομα στο σώμα μιας πρότασης. Συμπεραίνουμε επομένως, ότι ο αλγόριθμός μας είναι ορθός και πλήρης για τη Horn-$\mathcal{SHIQ}$. Αποδεικνύουμε ότι ο λογισμός τερματίζει.

Λήμμα 5.1.3. *Έστω ένα Horn-$\mathcal{SHIQ}$ TBox $\mathcal{T}$, ένα ABox, $\mathcal{A}$ και ένα ΣΕ, $\mathcal{Q}$. Όταν ο $\mathcal{I}_{\mathrm{HS}}$ εφαρμόζεται στο $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\}$ οι προτάσεις που παράγονται έχουν κάποιον από τους τύπους που φαίνονται στον Πίνακα 5.1 ή είναι ταυτολογίες.*

Απόδειξη. Οι προτάσεις που παράγονται από το λογισμό διαθέτουν συγκεκριμένους τύπους που παρουσιάζονται στον Πίνακα 5.1. Πράγματι, κάθε Horn-$\mathcal{SHIQ}$ αξίωμα που έχει μετατραπεί σε πρόταση λογικής πρώτης τάξης αντιστοιχίζεται σε κάποια από τις προτάσεις του Πίνακα 5.1. Οι ισχυρισμοί του $\mathcal{SHIQ}$ ABox, $\mathcal{A}$ αντιστοιχίζονται στους τύπους 7.1–8, και ο τύπος 1 περιγράφει κατά τα γνωστά ένα ΣΕ, $\mathcal{Q}$.

Όπως φαίνεται από τον Πίνακα 5.2 με τις δυνατές εφαρμογές κανόνων συμπερασμού του $\mathcal{I}_{\mathrm{HS}}$, οι προτάσεις που κατασκευάζονται είτε περιγράφονται από τους τύπους προτάσεων του Πίνακα 5.1, είτε είναι ταυτολογίες. □

Επομένως, λαμβάνοντας υπόψη το Λήμμα 5.1.3 και ότι τα $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\}$ περιλαμβάνουν ένα πεπερασμένο σύνολο συμβόλων, συμπεραίνουμε ότι ο $\mathcal{I}_{\mathrm{HS}}$ παράγει ένα πεπερασμένο πλήθος προτάσεων και άρα τερματίζει.

Σημειώνουμε, ότι ο Πίνακας 5.1 όπως είναι φανερό, επεκτείνει τον Πίνακα 2.4 ώστε να συμπεριληφθούν οι προτάσεις που προκύπτουν από την εκφραστικότητα της Horn-$\mathcal{SHIQ}$.

5.1.1 Κανόνες συμπερασμού με βασικές προτάσεις

Στην ενότητα αυτή θα δείξουμε ότι είναι δυνατό να διαχωρίσουμε τους κανόνες συμπερασμού που χρησιμοποιούν προτάσεις με μεταβλητές, από εκείνους που χρησιμοποιούν μόνο βασικές προτάσεις [70, 110]. Με άλλα λόγια, θα δείξουμε πως οι κανόνες του $\mathcal{I}_{\mathrm{HS}}$ είναι δυνατό να εφαρμοστούν εξαντλητικά σαν πρώτο βήμα, στις προτάσεις του σώματος ορολογίας, και σε δεύτερο βήμα, να ληφθούν υπόψη οι βασικές προτάσεις του σώματος ισχυρισμών.

Θεωρούμε τους κανόνες που εφαρμόζονται από το $\mathcal{I}_{\mathrm{HS}}$ και περιλαμβάνουν βασικές προτάσεις σα δευτερεύουσες προκείμενες, δηλαδή προτάσεις με τύπο 7.1, 7.2 και 8. Οι προτάσεις με τύπο 7.1, 7.2 συμμετέχουν σε κανόνες δυαδικής επίλυσης, ενώ η πρόταση με τύπο 8 συμμετέχει στον κανόνα υπέρθεσης. Συγκεκριμένα, μία προκείμενη με τύπο 7.1, είναι της μορφής $A(a)$, δηλαδή, δεν διαθέτει άτομα στο σώμα. Κάτι τέτοιο συμβαίνει επειδή από τον ορισμό της επίλυσης τα άτομα στην κεφαλή της δευτερεύουσας προκείμενης, πρέπει να είναι αυστηρώς μέγιστα. Μία δευτερεύουσα προκείμενη πρόταση με τύπο 8 είναι της μορφής $g(a) \approx b \leftarrow A(a)$, ή $a \approx b$. Πράγματι, σύμφωνα με τον ορισμό του κανόνα της θετικής υπέρθεσης, μία δευτερεύουσα προκείμενη με κεφαλή $a \approx b$ πρέπει να μην έχει άτομα στο σώμα της ώστε να είναι αυστηρά μέγιστη η κεφαλή. Τέλος, σημειώνουμε ότι οι κύριες προκείμενες δεν είναι δυνατό να περιλαμβάνουν συναρτήσεις στο σώμα τους, καθώς οι συναρτήσεις δεν ενοποιούνται με τις σταθερές των βασικών ατόμων. Διακρίνουμε τις περιπτώσεις εφαρμογής

Πίνακας 5.1: Τύποι Horn-$\mathcal{SHIQ}$ προτάσεων

1	$\mathcal{Q}(\vec{s}) \leftarrow D_j(\vec{t_j})$
2.1	$R(x, f(x)) \leftarrow \mathbf{A}(x) \wedge \mathbf{B}(f(x))$
2.2	$R(f(x), x) \leftarrow \mathbf{A}(x) \wedge \mathbf{B}(f(x))$
2.3	$A(f(x)) \leftarrow \mathbf{C}(x) \wedge \mathbf{B}(f(x))$
3.1	$R(x, y) \leftarrow P(x, y)$
3.2	$R(x, y) \leftarrow P(y, x)$
3.3	$A(x) \leftarrow \mathbf{C}(x) \wedge \mathbf{B}(f(x))$
4.1	$A(x) \leftarrow R(x, y) \wedge \mathbf{C}(y)$
4.2	$A(x) \leftarrow R(y, x) \wedge \mathbf{C}(y)$
5	$y \approx z \leftarrow \mathbf{A}(x) \wedge R(x, y) \wedge P(x, z) \wedge [B(y)] \wedge [C(z)]$
6.1	$f(x) \approx g(x) \leftarrow \mathbf{A}(x) \wedge [B(f(x))] \wedge [C(g(x))]$
6.2	$x \approx g(f(x)) \leftarrow \mathbf{A}(x) \wedge \mathbf{B}(f(x)) \wedge [C(g(f(x))])$
6.3	$t_1 \approx z \leftarrow \mathbf{A}(t_2) \wedge R(t_2, z) \wedge [B(t_1)] \wedge [C(z)]$
όπου οι όροι t_1, t_2 είναι $x, f(x)$, ή $f(x), x$, ή a, b	
7.1	$R(a, b) \leftarrow \mathbf{A}(a) \wedge \mathbf{B}(b)$
7.2	$A(a) \leftarrow \mathbf{B}(a) \wedge \mathbf{C}(b)$
8	$t_i \approx t_j \leftarrow \mathbf{A}(t_1) \wedge [B(t_2)] \wedge [C(t_3)]$
όπου ο όρος $t_{i,j}$, $i, j \in \{1, 2, 3\}$ είναι της μορφής a, Ή $f(a)$	
Το D_j στον τύπο 1 υποδηλώνει ένα κατηγόρημα έννοιας ή απλού ρόλου. Κάθε ρόλος στους τύπους $5-6.3$ μπορεί να αντικατασταθεί από τον ανάστροφό του. Για παράδειγμα, η πρόταση $y \approx z \leftarrow \mathbf{A}(x) \wedge R(y, x) \wedge R(x, z)$ έχει τύπο 5.	

των κανόνων της βασικής υπέρθεσης του $\mathcal{I}_{\mathrm{HS}}$, που χρησιμοποιούν ως δευτερεύουσα προκείμενη μία βασική πρόταση, με κριτήριο τον τύπο της κύριας προκείμενης:

- τύπος 1: ένας όρος με συνάρτηση στην κεφαλή πρότασης τύπου 1, δεν μπορεί παρά να έχει προέλθει από προηγούμενη αντικατάσταση και έτσι, με βάση τον ορισμό της βασικής υπέρθεσης, δεν είναι δυνατό να πραγματοποιηθεί υπέρθεση με κάποια πρόταση $f(a) \approx b \leftarrow A(a)$. Επιπλέον, η υπέρθεση με μία πρόταση $a \approx b$ υπονοεί ότι η κύρια προκείμενη τύπου 1 είναι βασική πρόταση με ένα μόνο άτομο. Εξετάζουμε, τώρα την περίπτωση του κανόνα επίλυσης με κύρια προκείμενη της μορφής $Q(\vec{s}) \leftarrow \bigwedge_i D_i(\vec{s_i})$, όπου ο όρος $\vec{s}_{(i)}$ δεν περιλαμβάνει συναρτησιακά σύμβολα, και δευτερεύουσες τις $7.1, 7.2$. Το συμπέρασμα είναι τύπου 1, περιλαμβάνει σταθερές και μπορεί να επιλυθεί μόνο με κάποια βασική πρόταση. Για παράδειγμα, θεωρούμε ένα ΣΕ $\mathcal{Q}$, $Q(\vec{s}) \leftarrow R(x, y) \wedge A(x) \wedge P(x, z)$ που επιλύεται με την $R(a, b)$ για να προκύψει το $\mathcal{Q}'$, $Q(\vec{s})\sigma \leftarrow (A(a) \wedge P(a, z))\sigma$, όπου $\sigma = \mathsf{MGU}(R(x, y), R(a, b))$. Όπως γίνεται κατανοητό, η $\mathcal{Q}'$ περιλαμβάνει σταθερές καθώς και μεταβλητές και μπορεί να επιλυθεί μόνο με κάποια βασική πρόταση.

- τύποι 2.1, 2.2, 2.3: οι προτάσεις της μορφής $R(x, f(x)) \leftarrow B(x), R(f(x), x) \leftarrow B(x), A(f(x)) \leftarrow B(x)$ όπου ο όρος $f(x)$ δεν έχει προέλθει από αντικατάσταση

Πίνακας 5.2: Κανόνες του $\mathcal{I}_{\mathrm{HS}}$ σε Horn-$\mathcal{SHIQ}$ προτάσεις με ισότητα.

$$\mathbf{2.1(2.2)+6.1=2.1(2.2)}\quad \frac{R(x,f(x)) \leftarrow A(x) \quad f(x) \approx g(x) \leftarrow B(x)}{R(x,g(x)) \leftarrow A(x) \wedge B(x)}$$

$$\mathbf{2.3+2.3=2.3}\quad \frac{A(f(x)) \leftarrow B(f(x)) \wedge C(x) \quad B(f(x)) \leftarrow D(x)}{A(f(x)) \leftarrow C(x) \wedge D(x)}$$

$$\mathbf{2.1(2.2)+6.2=2.2(2.1)}\quad \frac{R(x,f(x)) \leftarrow A(x) \quad f(g(x)) \approx x \leftarrow B(x)}{R(g(x),x) \leftarrow A(g(x)) \wedge B(x)}$$

$$\mathbf{2.3+6.1=2.3}\quad \frac{A(f(x)) \leftarrow B(x) \quad f(x) \approx g(x) \leftarrow C(x)}{A(g(x)) \leftarrow B(x) \wedge C(x)}$$

$$\mathbf{2.1(2.2)+2.3=2.1(2.2)}\quad \frac{R(x,f(x)) \leftarrow A(x) \wedge B(f(x)) \quad B(f(x)) \leftarrow C(x)}{R(x,f(x)) \leftarrow A(x) \wedge C(x)}$$

$$\mathbf{2.3+6.2=3.3}\quad \frac{A(f(x)) \leftarrow B(x) \quad f(g(x)) \approx x \leftarrow C(x)}{A(x) \leftarrow B(g(x)) \wedge C(x)}$$

$$\mathbf{5+2.1=6.3}\quad \frac{y \approx z \leftarrow \mathbf{A}(x) \wedge R(x,y) \wedge R(x,z) \wedge B(y) \wedge B(z) \quad R(x,f(x)) \leftarrow \mathbf{C}(x)}{f(x) \approx z \leftarrow \mathbf{A}(x) \wedge \mathbf{C}(x) \wedge R(x,z) \wedge B(f(x)) \wedge B(z)}$$

$$\mathbf{5+2.2=6.3}\quad \frac{y \approx z \leftarrow \mathbf{A}(x) \wedge R(x,y) \wedge R(x,z) \wedge B(y) \wedge B(z) \quad R(f(x),x) \leftarrow \mathbf{C}(x)}{x \approx z \leftarrow A(f(x)) \wedge \mathbf{C}(x) \wedge R(f(x),z) \wedge B(x) \wedge B(z)}$$

$$\mathbf{6.1+2.3=6.1}\quad \frac{f(x) \approx g(x) \leftarrow \mathbf{A}(x) \wedge B(g(x)) \quad B(g(x)) \leftarrow \mathbf{C}(x)}{f(x) \approx g(x) \leftarrow \mathbf{A}(x) \wedge \mathbf{C}(x)}$$

$$\mathbf{6.2+2.3=6.2}\quad \frac{f(g(x)) \approx x \leftarrow \mathbf{A}(x) \wedge B(f(g(x))) \wedge C(g(x)) \quad B(f(x)) \leftarrow \mathbf{D}(x)}{f(g(x)) \approx x \leftarrow \mathbf{A}(x) \wedge \mathbf{D}(g(x)) \wedge C(g(x))}$$

$$\mathbf{6.3+2.1=6.1}\quad \frac{f(x) \approx z \leftarrow \mathbf{A}(x) \wedge R(x,z) \wedge B(z) \quad R(x,g(x)) \leftarrow \mathbf{C}(x)}{f(x) \approx g(x) \leftarrow \mathbf{A}(x) \wedge \mathbf{C}(x) \wedge B(g(x))}$$

$$\mathbf{6.3+2.1=6.2}\quad \frac{x \approx z \leftarrow A(f(x)) \wedge R(f(x),z) \wedge B(x) \wedge C(z) \quad R(x,g(x)) \leftarrow \mathbf{D}(x)}{g(f(x)) \approx x \leftarrow \mathbf{A}(f(x)) \wedge B(x) \wedge C(g(f(x))) \wedge D(f(x))}$$

$$\mathbf{6.3+2.2=6.2}\quad \frac{f(x) \approx z \leftarrow \mathbf{A}(x) \wedge R(x,z) \wedge B(z) \quad R(g(x),x) \leftarrow \mathbf{C}(x)}{f(g(x)) \approx x \leftarrow \mathbf{A}(g(x)) \wedge \mathbf{C}(x) \wedge B(x)}$$

$$\mathbf{6.3+2.3=6.3}\quad \frac{f(x) \approx z \leftarrow \mathbf{A}(x) \wedge R(x,z) \wedge B(z) \wedge D(f(x)) \quad D(f(x)) \leftarrow \mathbf{C}(x)}{f(x) \approx z \leftarrow \mathbf{A}(x) \wedge R(x,z) \wedge B(z) \wedge \mathbf{C}(x)}$$

$$\frac{x \approx z \leftarrow A(f(x)) \wedge R(f(x),z) \wedge B(x) \wedge C(z) \quad A(f(x)) \leftarrow \mathbf{D}(x)}{x \approx z \leftarrow \mathbf{D}(x) \wedge R(f(x),z) \wedge B(x) \wedge C(z)}$$

$$\mathbf{6.3+2.2}\quad \frac{x \approx z \leftarrow A(f(x)) \wedge R(f(x),z) \wedge B(x) \wedge C(z) \quad R(f(x),x) \leftarrow \mathbf{D}(x)}{x \approx x \leftarrow A(f(x)) \wedge B(x) \wedge C(x) \wedge \mathbf{D}(x)}$$

$$\mathbf{6.3+2.1=8}\quad \frac{c \approx z \leftarrow \mathbf{A}(a) \wedge R(a,z) \wedge B(c) \wedge C(z) \quad R(x,f(x)) \leftarrow \mathbf{D}(x)}{f(a) \approx b \leftarrow \mathbf{A}(a) \wedge \mathbf{D}(a) \wedge B(c) \wedge C(f(a))}$$

$$\mathbf{2.1+8=7.1}\quad \frac{R(x,f(x)) \leftarrow C(x) \quad f(a) \approx b \leftarrow A(a) \wedge B(b)}{R(a,b) \leftarrow \mathbf{A}(a) \wedge B(a)}$$

$$\mathbf{8+7.1=8}\quad \frac{a \approx b \leftarrow A(a) \wedge B(b) \wedge C(c) \quad A(a)}{a \approx b \leftarrow B(b) \wedge C(c)}$$

$$\mathbf{5+7.1=6.3}\quad \frac{y \approx z \leftarrow \mathbf{A}(x) \wedge R(x,y) \wedge R(x,z) \wedge B(y) \wedge B(z) \quad R(a,b) \leftarrow \mathbf{C}(a) \wedge \mathbf{D}(b)}{b \approx z \leftarrow \mathbf{A}(a) \wedge \mathbf{C}(a) \wedge R(a,z) \wedge B(b) \wedge \mathbf{D}(b) \wedge B(z)}$$

$$\mathbf{6.3+7.1=8}\quad \frac{f(x) \approx z \leftarrow \mathbf{A}(x) \wedge R(x,z) \wedge B(z) \quad R(a,b) \leftarrow \mathbf{C}(a) \wedge \mathbf{D}(b)}{f(a) \approx b \leftarrow \mathbf{A}(a) \wedge \mathbf{C}(a) \wedge B(b) \wedge \mathbf{D}(b)}$$

$$\mathbf{6.3+7.1=8}\quad \frac{c \approx z \leftarrow \mathbf{A}(a) \wedge R(a,z) \wedge B(c) \wedge C(z) \quad R(a,b) \leftarrow \mathbf{C}(a) \wedge \mathbf{D}(b)}{c \approx b \leftarrow \mathbf{A}(a) \wedge \mathbf{C}(a) \wedge C(b) \wedge \mathbf{D}(b)}$$

σε κάποιο προηγούμενο βήμα, συμμετέχουν ως κύριες σε κανόνα βασικής υπέρθεσης με πρόταση τύπου 8, της μορφής $f(a) \approx b \leftarrow C(a)$ για να προκύψουν προτάσεις με τύπο 7.1, ή 7.2.

- τύποι 3.1, 3.2: επιλύονται με μία πρόταση με τύπο 7.1, και παράγεται ακόμα μία πρόταση με τύπο 7.1.

- τύπος 3.3: μία πρόταση της μορφής $A(x) \leftarrow B(x)$, επιλύεται με μία πρόταση 7.2, παράγοντας μία βασική πρόταση με τύπο 7.2.

- τύποι 4.1, 4.2: στις προτάσεις αυτές επιλέγεται πάντοτε ο ρόλος και επιλύεται με μία πρόταση 7.1 παράγοντας ένα συμπέρασμα με τύπο 7.2.

- τύπος 5: μία πρόταση της μορφής $y \approx z \leftarrow A(x) \wedge R(x,y) \wedge R(x,z) \wedge B(y) \wedge B(z)$ επιλύεται με πρόταση 7.1, της μορφής $R(a,b) \leftarrow C(a) \wedge D(b)$, για να παράγει μία πρόταση με τύπο 6.3, $a \approx z \leftarrow \mathbf{A}(a) \wedge R(a,z) \wedge \mathbf{B}(b) \wedge B(z)$, που περιλαμβάνει σταθερές και μπορεί μόνο να επιλυθεί με ένα τύπο 7.2.

- τύπος 6.1: εαν είναι της μορφής $f(x) \approx g(x) \leftarrow A(x)$ δε συμμετέχει σαν κύρια προκείμενη σε κανόνα υπέρθεσης καθώς οι όροι $f(x), g(x)$ έχουν εισαχθεί στην πρόταση από προηγούμενο βήμα ενοποίησης. Επιπλέον, δεν μπορεί να συμμετέχει σε κανόνα επίλυσης, αφού η κεφαλή της είναι μέγιστη.

- τύπος 6.2: εαν είναι της μορφής $f(g(x)) \approx x \leftarrow A(x)$, ο μέγιστος όρος $f(g(x))$ δεν ενοποιείται με βασικό όρο. Επιπλέον, δε συμμετέχει σε κανόνα επίλυσης καθώς δε διαθέτει επιλεγμένο άτομο.

- τύπος 6.3: προτάσεις της μορφής $f(x) \approx x \leftarrow A(x) \wedge R(x,z) \wedge B(z)$, ή $b \approx z \leftarrow A(a) \wedge R(a,x) \wedge B(z)$ επιλύονται με 7.1, $R(a,b) \leftarrow A(a) \wedge B(b)$, για να προκύψει κάποιος τύπος 8 της μορφής $f(a) \approx b \leftarrow A(a) \wedge B(b)$, ή $a \approx b \leftarrow A(a) \wedge B(b)$.

- τύπος 7.1, 7.2: προτάσεις της μορφής $B(a) \leftarrow A(a) \wedge C(b)$, επιλύονται με προτάσεις 7.2 της μορφής $A(a)$, για να προκύψει πρόταση με τύπο 7.2. Μία πρόταση της μορφής $R(a,b) \leftarrow A(a) \wedge D(b)$ δε συμμετέχει ως κύρια προκείμενη σε κανόνα επίλυσης, καθώς το $R(a,b)$ στην κεφαλή είναι αυστηρά μέγιστο.

- τύπος 8: μία πρόταση της μορφής $a \approx b \leftarrow A(a) \wedge B(b)$ επιλύεται με πρόταση τύπου 7.2, της μορφής $A(a)$. Προκύπτει μία βασική πρόταση με τύπο 8, $a \approx b \leftarrow B(b)$. Η πρόταση $f(a) \approx b \leftarrow \mathbf{A}(a)$ δε συμμετέχει ως κύρια προκείμενη στον κανόνα της επίλυσης, καθώς η κεφαλή της είναι μέγιστη.

Ορισμός 5.1.4. Έστω $\mathcal{T}$ ένα Horn-$\mathcal{SHIQ}$ TBox, $\mathcal{A}$ ένα ABox και $\mathcal{Q}$ ένα ΣΕ. Το σύνολο $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\})$ περιλαμβάνει τις προτάσεις με τύπο 2.1–2.3 που ανήκουν στο $\mathcal{T}$, τις προτάσεις με τύπο $1, 3.1–5, 6.3$ που δεν περιλαμβάνουν συναρτήσεις στο σώμα τους και προέκυψαν εφαρμόζοντας εξαντλητικά τους κανόνες του $\mathcal{I}_{\mathrm{H}\mathcal{S}}$ στο $\mathcal{T} \cup \{\mathcal{Q}\}$. $\triangle$

Το Λήμμα 5.1.5 προκύπτει απ'ευθείας από την παραπάνω ανάλυση.

Λήμμα 5.1.5. *Έστω $\mathcal{T}$ ένα Horn-$\mathcal{SHIQ}$ TBox, $\mathcal{A}$ ένα ABox και $\mathcal{Q}$ ένα ΣΕ. Μία πρόταση $\mathcal{C}$ που παράγεται από ένα κανόνα συμπερασμού $\mathcal{C}_1, \mathcal{C}_2 \vdash^{\mathcal{I}_{\mathrm{HS}}} \mathcal{C}$, όπου η $\mathcal{C}_2$ είναι βασική πρόταση, μπορεί είναι βασική ή μπορεί να επιλυθεί μόνο με μία βασική πρόταση. Επιπλέον, κάθε πρόταση $\mathcal{C}_1$ που δεν περιλαμβάνει σταθερές ανήκει στο $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\})$.*

Λήμμα 5.1.6. *Έστω $\mathcal{T}$ ένα Horn-$\mathcal{SHIQ}$ TBox, $\mathcal{A}$ ένα ABox, και $\mathcal{Q}$ ένα ΣΕ. Το $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\}$ είναι μη ικανοποιήσιμο ανν το $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$ είναι μη ικανοποιήσιμο.*

Απόδειξη. Εφόσον ο $\mathcal{I}_{\mathrm{HS}}$ είναι ορθός και πλήρης λογισμός υπέρθεσης, το σύνολο $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\}$ είναι μη ικανοποιήσιμο αν συνεπάγεται την κενή πρόταση, $\perp$. Θεωρούμε την παραγωγή $\mathcal{C}_0, \mathcal{C}_1, \ldots, \mathcal{C}_m$ μιας πρότασης $\mathcal{C}_m$ από το $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\}$, τέτοια ώστε $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\} \cup \{\mathcal{C}_1, \ldots, \mathcal{C}_m\} = \mathsf{Sat}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$, όπου το σύνολο $\mathsf{Sat}(\mathcal{T} \cup \{\mathcal{Q}\})$ υποδηλώνει τον κορεσμό του $\mathcal{T} \cup \{\mathcal{Q}\}$ με βάση το $\mathcal{I}_{\mathrm{HS}}$. Οι κανόνες που εφαρμόζονται για την κατασκευή της παραγωγής $\mathcal{C}_0, \mathcal{C}_1, \ldots, \mathcal{C}_m$ δεν εμπλέκουν βασικές προτάσεις από το $\mathcal{A}$. Προκειμένου να προκύψει το σύνολο $\mathsf{Sat}(\mathcal{T} \cup \{\mathcal{Q}\})$, έχουν ήδη εφαρμοστεί εξαντλητικά οι κανόνες συμπερασμού σε προτάσεις με τύπο διαφορετικό από 7.1, 7.2, και 8. Επομένως, κάθε πρόταση $\mathcal{C}_i$, με $i > m$, παράγεται επιλύοντας την $\mathcal{C}_{i-1}$ με κάποια βασική πρόταση. Από το Λήμμα 5.1.5 προκύπτει ότι το σύνολο $\mathsf{Sat}(\mathcal{T} \cup \{\mathcal{Q}\})$ μπορεί να αντικατασταθεί από το $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\})$. □

Λήμμα 5.1.7. *Έστω $\mathcal{T}$, ένα Horn-$\mathcal{SHIQ}$ TBox, $\mathcal{A}$, ένα ABox. Έστω $\mathcal{Q}$, ένα ΣΕ με κατηγόρημα ερωτήματος $Q(\vec{s})$. Ισχύει ότι $\vec{a} \in \mathsf{cert}(\mathcal{Q}, \mathcal{T}, \mathcal{A})$ ανν το $Q(\vec{a})$ παράγεται απο το $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\}$ με βάση τον $\mathcal{I}_{\mathrm{HS}}$.*

Απόδειξη. Από τον ορισμό των απαντήσεων ΣΕ ισχύει ότι $\vec{a} \in \mathsf{cert}(\mathcal{Q}, \mathcal{T}, \mathcal{A})$ ανν το σύνολο προτάσεων $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\} \cup \{\perp \leftarrow Q(\vec{a})\}$ είναι ικανοποιήσιμο. Θα δείξουμε ότι το σύνολο $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\} \cup \{\perp \leftarrow Q(\vec{a})\}$ είναι μη ικανοποιήσιμο ανν η πρόταση $Q(\vec{a})$ παράγεται από το $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\}$ μέσω $\mathcal{I}_{\mathrm{HS}}$.

Έστω ότι το σύνολο $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\} \cup \{\perp \leftarrow Q(\vec{a})\}$ είναι μη ικανοποιήσιμο. Ο κορεσμός του $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\}$ με βάση το $\mathcal{I}_{\mathrm{HS}}$ δεν μπορεί να παράγει την κενή πρόταση εφόσον δε θεωρούμε μη συνεπείς Horn-$\mathcal{SHIQ}$ οντολογίες. Επιπλέον, το κατηγόρημα Q δεν εμφανίζεται στο σώμα κάποιας πρότασης. Επομένως, προκειμένου να εξάγουμε την κενή πρόταση από το $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\} \cup \{\perp \leftarrow Q(\vec{a})\}$ μέσω $\mathcal{I}_{\mathrm{HS}}$, το $Q(\vec{a})$ πρέπει να συμπεριλαμβάνεται στον κορεσμό του $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\}$, εφόσον ο λογισμός της βασικής υπέρθεσης είναι ορθός και πλήρης.

Αντίστροφα, αν η $\mathcal{Q}(\vec{a})$ παράγεται από το $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\}$ μέσω $\mathcal{I}_{\mathrm{HS}}$, τότε το $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\} \cup \{\perp \leftarrow Q(\vec{a})\}$ είναι μη ικανοποιήσιμο. □

Από τα παραπάνω λήμματα 5.1.6, 5.1.7 προκύπτει το παρακάτω θεώρημα.

Θεώρημα 5.1.8. *Έστω $\mathcal{T}$ ένα Horn-$\mathcal{SHIQ}$ TBox, $\mathcal{A}$ ένα ABox και ένα ΣΕ $\mathcal{Q}$, με κατηγόρημα ερωτήματος $Q(\vec{s})$. Ισχύει ότι $\vec{a} \in \mathsf{ans}(\mathcal{Q}, \mathcal{T} \cup \mathcal{A})$ ανν το $Q(\vec{a})$ παράγεται από το $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$ με τον $\mathcal{I}_{\mathrm{HS}}$.*

Απομένει να δείξουμε ότι οι προτάσεις στο $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\})$ μπορούν να μετασχηματισούν σε κανόνες datalog. Γενικά, για το σκοπό αυτό παραλείπουμε τις προτάσεις με συναρτησιακά σύμβολα, αλλά καθώς μερικές από αυτές αλληλεπιδρούν με βασικές προτάσεις ορίζουμε μία αντιστοίχιση, όμοια με αυτή που παρουσιάζεται στο [70].

Ορισμός 5.1.9. Για μία σταθερά a, μία μεταβλητή x, και ένα συναρτησιακό σύμβολο f, ο τελεστής λ αντιστοιχίζει όρους σε όρους με τον εξής τρόπο:

- $\lambda(a) = a$
- $\lambda(f(a)) = a_f$, όπου το a_f είναι μία νέα μοναδική σταθερά
- $\lambda(x) = x$
- $\lambda(f(x)) = x_f$, όπου το x_f είναι μία νέα μοναδική μεταβλητή

Ο τελεστής λ αντιστοιχίζει μία Horn-$\mathcal{SHIQ}$ πρόταση $\mathcal{C}$ σε μία πρόταση $\lambda(\mathcal{C})$ που δεν περιέχει συναρτήσεις με τον παρακάτω τρόπο:

- κάθε όρος t στο $\mathcal{C}$ αντικαθίσταται με $\lambda(t)$
- αν μία μεταβλητή x_f εισάγεται στο $\lambda(\mathcal{C})$ τότε ένα άτομο $S_f(x, x_f)$ εισάγεται στο σώμα της $\lambda(\mathcal{C})$

$\triangle$

Θεωρούμε ένα Horn-$\mathcal{SHIQ}$ TBox $\mathcal{T}$, $\mathcal{A}$ ένα ABox και ένα ΣΕ $\mathcal{Q}$. Όπως έχουμε ήδη αναφέρει, προκειμένου να κατασκευάσουμε ένα datalog πρόγραμμα ισοδύναμο με το $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\}$, δεν μπορούμε να αγνοήσουμε κάθε πρόταση στο $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\})$ που περιλαμβάνει συναρτήσεις. Οι προτάσεις με συναρτήσεις που αλληλεπιδούν με βασικές προτάσεις του $\mathcal{A}$, δηλαδή οι προτάσεις με τύπο 6.3 που δεν περιλαμβάνουν συναρτήσεις στο σώμα και με τύπο $2.1-2.3$ που ανήκουν στο $\mathcal{T}$, συμπεριλαμβάνονται στο datalog προγράμμα εφαρμόζοντας τον τελεστή του Ορισμού 5.1.9.

Ορισμός 5.1.10. Έστω ένα Horn-$\mathcal{SHIQ}$ TBox $\mathcal{T}$, ένα Horn-$\mathcal{SHIQ}$ ABox $\mathcal{A}$, και ένα ΣΕ $\mathcal{Q}$. Το σύνολο $\mathsf{DD}(\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A})$ είναι ένα datalog πρόγραμμα που περιλαμβάνει τους κανόνες που προκύπτουν εφαρμόζοντας τον τελεστή λ σε κάθε πρόταση του $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\})$, καθώς και την πρόταση $S_f(a, a_f)$ για κάθε άτομο $S_f(x, x_f)$ και κάθε σταθερά a στο $\mathcal{A}$. $\triangle$

Λήμμα 5.1.11. *Έστω ένα Horn-$\mathcal{SHIQ}$ TBox $\mathcal{T}$, ένα Horn-$\mathcal{SHIQ}$ ABox $\mathcal{A}$, και ένα ΣΕ $\mathcal{Q}$. Το $\mathcal{T} \cup \mathcal{A} \cup \{\mathcal{Q}\}$ είναι μη ικανοποιήσιμο αν το iff $\mathsf{DD}(\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A})$ είναι μη ικανοποιήσιμο.*

Απόδειξη. ($\Rightarrow$) Έστω μία πρόταση $\mathcal{C}_i$ που παράγεται από το $\mathsf{DD}(\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A})$ σε βάθος i. Θα δείξουμε ότι για κάθε βασική πρόταση $\mathcal{C}_i$ που δεν περιλαμβάνει άτομα της μορφής $S(a, a_f), A(a_f)$ υπάρχει μία πρόταση $\mathcal{C}'_j, j \leq i$ που προκύπτει από το $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$ έτσι ώστε $\lambda(\mathcal{C}'_j) = \mathcal{C}_i$. Αποδεικνύουμε τον ισχυρισμό με επαγωγή στο βάθος i της παραγωγής από το $\mathsf{DD}(\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A})$.
Βασική περίπτωση: για $i = 0$ ο ισχυρισμός ισχύει από τον ορισμό του DD.
Επαγωγικό βήμα: υποθέτουμε ότι για κάθε βασική πρόταση $C_i, i \leq k$ που παράγεται από το $\mathsf{DD}(\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A})$, σε βάθος το πολύ k, που δεν περιλαμβάνει άτομα της μορφής $S(a, a_f), A(a_f)$ υπάρχει κάποια πρόταση $\mathcal{C}'_j$, $j \leq k$ στο $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$ ώστε $\lambda(\mathcal{C}'_j) = \mathcal{C}_i$ (ΕΥ1). Θεωρούμε τους πιθανούς κανόνες συμπερασμού που εφαρμόζονται στην $\mathcal{C}_k$:

1. Η $\mathcal{C}_k$ συμμετέχει σε ένα κανόνα υπέρθεσης με δευτερεύουσες προκείμενες $\mathcal{C}_{sp}$, της μορφής $a \approx b$, ή $a_f \approx b$. Από την ΕΥ1 υπάρχει κάποια πρόταση $\mathcal{C}'_{sp}$ που προκύπτει από το $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$ τέτοια ώστε $\lambda(\mathcal{C}'_{sp}) = \mathcal{C}_{sp}$, και επομένως η $\mathcal{C}'_{sp}$ είναι της μορφής $a \approx b$, ή $f(a) \approx b$. Αν θεωρήσουμε ότι η $\mathcal{C}_k$ είναι η πρόταση $R(a,b) \leftarrow B(a)$, ή η $A(a)$, τότε επιλύεται με κάποια $a \approx b$ παράγοντας την $\mathcal{C}_{k+1}$. Από την ΕΥ1 ισχύει ότι μία πρόταση $\mathcal{C}'_m$ παράγεται από το $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$ ώστε $\lambda(\mathcal{C}'_m) = \mathcal{C}_k$. Προφανώς, η $\mathcal{C}'_m$ είναι της μορφής $R(a,b) \leftarrow B(a)$ ή $A(a)$, οπότε και η επίλυση με την $a \approx b$ παράγει την $\mathcal{C}'_{m+1}$, τέτοια ώστε $\lambda(\mathcal{C}'_{m+1}) = \mathcal{C}_{k+1}$. Στη συνέχεια, αν θεωρήσουμε ότι η $\mathcal{C}_k$ είναι της μορφής $R(a,a_f) \leftarrow B(a)$, ή $A(a_f)$ τότε η υπέρθεση με την $a_f \approx b$ παράγει την $\mathcal{C}_{k+1}$ της μορφής $R(a,b) \leftarrow B(a)$, ή $A(b)$. Η πρόταση $\mathcal{C}_k$ υπονοεί την ύπαρξη κάποιας $\mathcal{C}_1$ στο $\mathsf{DD}(\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A})$ της μορφής $R_1(x,x_f) \leftarrow B(x) \wedge S(x,x_f)$. Η $\mathcal{C}_1$ επιλύεται με την $S_f(a,a_f)$ που περιλαμβάνεται στο $\mathsf{DD}(\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A})$, συμπεραίνοντας την $R_1(a,a_f) \leftarrow B(a)$. Η πρόταση C_k κατασκευάζεται από διαδοχικές εφαρμογές κανόνων συμπερασμού της μορφής $R_{i+1}(x,y) \leftarrow R_i(x,y), R_i(a,a_f) \leftarrow B(a) \vdash R_{i+1}(a,a_f) \leftarrow B(a)$. Εξ' ορισμού το σύνολο DD $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$ περιλαμβάνει τις προτάσεις $R_1(x,f(x)) \leftarrow B(x)$ και $R_{i+1}(x,y) \leftarrow R_i(x,y)$. Επομένως, η $R(x,f(x)) \leftarrow B(x)$ παράγεται από διαδοχικές εφαρμογές κανόνων της μορφής $R_{i+1}(x,y) \leftarrow R_i(x,y), R_i(x,f(x)) \leftarrow B(x) \vdash R_{i+1}(x,f(x)) \leftarrow B(x)$. Η πρόταση $R(x,f(x)) \leftarrow B(x)$ επιλύεται με την $\mathcal{C}'_{sp} = f(a) \approx b$ συμπεραίνοντας την $R(a,b) \leftarrow B(a)$.

2. Η $\mathcal{C}_k$ συμμετέχει σε κανόνα επίλυσης. Διακρίνουμε διαφορετικές περιπτώσεις ανάλογα με τη μορφή της $\mathcal{C}_k$:

   - Η $\mathcal{C}_k$ παράγεται από μία πρόταση $\mathcal{C}_1$ τέτοια ώστε $\mathcal{C}_1 = \lambda(\mathcal{C}'_1)$ όπου η $\mathcal{C}'_1$ περιλαμβάνεται στο $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$ και έχει τύπο 1, 4.1, 4.2, ή 5. Σύμφωνα με τον ορισμό του DD, ισχύει ότι $\mathcal{C}_1 = \mathcal{C}'_1$. Θεωρούμε τους κανόνες $\mathcal{C}_i, \mathcal{C}_{sp_i} \vdash \mathcal{C}_{i+1}$, $1 \leq i \leq l$, όπου οι δευτερεύουσες προκείμενες $\mathcal{C}_{sp_i}$ είναι της μορφής $R(a,b) \leftarrow A(a)$, δηλαδή δεν περιλαμβάνουν όρους a_f. Από την ΕΥ1, οι $\mathcal{C}_{sp_i}$ προκύπτουν επίσης από το $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$. Προφανώς, οι προτάσεις $\mathcal{C}_{i+1}$ παράγονται από το $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$. Εαν η πρόταση $\mathcal{C}_l$ δεν περιλαμβάνει μεταβλητές τότε έχουμε αποδείξει τον ισχυρισμό.

     Εν συνεχεία, θεωρούμε την περίπτωση που η $\mathcal{C}_l$ περιέχει μεταβλητές, δεν είναι δηλαδή βασική πρόταση, και επιλύεται με κάποια πρόταση της μορφής $R(a,a_f) \leftarrow B(a), A(a_f)$. Η $\mathcal{C}_l$ πρέπει να περιλαμβάνει ένα άτομο $R(s,x), A(s)$ όπου ο όρος s είναι είτε σταθερά a, είτε μεταβλητή. Από την επίλυση που κάναμε παραπάνω για την Περίπτωση 1, προκύπτει ότι για κάθε πρόταση $R(a,a_f) \leftarrow B(a)$, ή $A(a_f)$ που παράγεται από το $\mathsf{DD}(\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A})$ υπάρχει κάποια $R(x,f(x)) \leftarrow B(x)$, ή $A(f(x)) \leftarrow B(x)$ που παράγεται από το $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$. Οι προτάσεις $R(x,f(x)) \leftarrow B(x)$, $A(f(x)) \leftarrow B(x)$ επιλύονται με την $\mathcal{C}_l$ στα άτομα $R(s,x), A(s)$, με αποτέλεσμα μία πρόταση $\mathcal{C}'_{l+1}$. Σημειώνουμε ότι αν η $\mathcal{C}_{l+1}$ περιλαμβάνει κάποιο άτομο $S(t,a_f)$, ή $C(a_f)$, τότε η πρόταση $\mathcal{C}'_{l+1}$ περιλαμβάνει κάποιο $S(t,f(s))$, ή $C(f(s))$. (Προκειμένου να παράγουμε μία βασική πρόταση

από την $\mathcal{C}_l$ η οποία δεν περέχει άτομα της μορφής $S(a, a_f)$, ή $C(a_f)$, είναι απαραίτητο η $\mathcal{C}_{l+1}$ να επιλυθεί με κάποια πρόταση $S(a, a_f) \leftarrow B(a)$, ή $C(a_f)$.) Θεωρούμε τους κανόνες $\mathcal{C}_j, \mathcal{C}_{sp_j} \vdash \mathcal{C}_{j+1}, l \leq j \leq m$ που παράγουν μία βασική πρόταση $\mathcal{C}_m$ που δεν περιλαμβάνει άτομα $S(a, a_f)$, ή $A(a_f)$. Μπορούμε να κατασκευάσουμε μία παραγωγή $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$, $\mathcal{C}'_j, \mathcal{C}'_{sp_j} \vdash \mathcal{C}'_{j+1}, l \leq j \leq m$ τέτοια ώστε για κάθε $\mathcal{C}_{sp_j}$ της μορφής $S(a, a_f) \leftarrow B(a)$, η $\mathcal{C}'_{sp_j}$ είναι της μορφής $S(x, f(x)) \leftarrow B(x)$. Έτσι, η $\mathcal{C}'_j$ διαφοροποιείται από την $\mathcal{C}_j$ στα άτομα $S(t, a_f)$ που περιλαμβάνει η τελευταία, ενώ στη θέση τους η $\mathcal{C}'_j$ περιλαμβάνει τα $S(t, f(s))$. Επιπλέον, αφού η $\mathcal{C}_m$ δεν περιέχει άτομα με όρους a_f προκύπτει ότι η $\mathcal{C}_m$ είναι χωρίς συναρτήσεις. Έστω $D_1(a_1), D_2(a_2) \ldots$ τα άτομα στο σώμα της $\mathcal{C}_m$, όπου οι όροι $a_1, a_2, \ldots$ είναι σταθερές. Εάν οι προτάσεις $D_i(a_i)$ παράγονται από το DD($\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$), τότε από την επαγωγική υπόθεση (ΕΥ1), προκύπτει ότι παράγονται και από το $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$. Εφόσον η $\mathcal{C}'_m$ δεν περιλαμβάνει συναρτήσεις τα άτομα στο σώμα της επιλύονται με τις $D_i(a_i)$ ώστε να προκύψει μία $\mathcal{C}'_g$, τέτοια ώστε $\lambda(\mathcal{C}'_g) = \mathcal{C}_g$, όπου η $\mathcal{C}_g$ προκύπτει από το $\mathcal{C}_m$.

- Η $\mathcal{C}_k$ είναι της μορφής $R(x, x_f) \leftarrow A(x) \wedge S_f(x, x_f)$ επιλύεται με $S_f(a, a_f)$ για να προκύψει η $R(a, a_f) \leftarrow A(a)$, που όμως περιλαμβάνει τον a_f.

- Η $\mathcal{C}_k$ είναι της μορφής $P(x, y) \leftarrow R(x, y)$ (ή $P(y, x) \leftarrow R(x, y)$). Από τον ορισμό του DD, η $\mathcal{C}_k$ περιέχεται στο $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$. Η $\mathcal{C}_k$ επιλύεται με την $R(a, b) \leftarrow B(a)$, ή $R(a, a_f) \leftarrow B(a)$ με αποτέλεσμα την $P(a, b) \leftarrow B(a)$, ή την $P(a, a_f) \leftarrow B(a)$, αντίστοιχα. Εστιάζουμε το ενδιαφέρον μας στη $P(a, b) \leftarrow B(a)$ που δεν περιέχει τον όρο a_f. Από την επαγωγική υπόθεση η $R(a, b) \leftarrow B(a)$ προκύπτει από το $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$, και έτσι, η $P(a, b) \leftarrow B(a)$ προκύπτει επίσης από το $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$.

- Η $\mathcal{C}_k$ παράγεται από την πρόταση $\mathcal{C}_1$ της μορφής $x_f \approx z \leftarrow A(x) \wedge R(x, z) \wedge S_f(x, x_f) \wedge B(z)$. Από τον ορισμό του DD υπάρχει μία βασική πρόταση $S_f(a, a_f)$ στο DD($\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$). Η επίλυση της $\mathcal{C}_1$ με την $S_f(a, a_f)$ παράγει τη $\mathcal{C}_2$ της μορφής $a_f \approx z \leftarrow A(a) \wedge R(a, z) \wedge B(b)$. Προκειμένου να παράγουμε μία βασική πρόταση που δεν αναφέρει τα $S(a, a_f), A(a_f)$, η $\mathcal{C}_2$ πρέπει να επιλυθεί με μία πρόταση $R(a, b) \leftarrow B(a)$ για να παράγουμε την $\mathcal{C}_3$, της μορφής $a_f \approx b \leftarrow A(a) \wedge B(b)$. Επίσης, η $R(a, b)$ είναι δυνατό να επιλυθεί με την $\mathcal{C}_1$ συμπεραίνοντας την $\mathcal{C}_2^*$ της μορφής $x_f \approx b \leftarrow A(a) \wedge S_f(a, x_f) \wedge B(a)$, η οποία επιλύεται με την $S_f(a, a_f)$ για να παραχθεί η $\mathcal{C}_3$. Από την επαγωγική υπόθεση, υπάρχει μία πρόταση $\mathcal{C}'_{sp}$ τέτοια ώστε $\mathcal{C}'_{sp} = \lambda(R(a, b) \leftarrow B(a))$. Προφανώς, η $\mathcal{C}'_{sp}$ είναι της μορφής $R(a, b) \leftarrow B(a)$. Επίσης από τον ορισμό του DD υπάχρει μία πρόταση $\mathcal{C}'_1$ της μορφής $f(x) \approx z \leftarrow A(x) \wedge R(x, z) \wedge B(z)$, $\lambda(\mathcal{C}'_1) = \mathcal{C}_1$. Ο κανόνας της επίλυσης στις προτάσεις $\mathcal{C}'_1, \mathcal{C}'_{sp}$ παράγει την $\mathcal{C}'_2$, $f(a) \approx b \leftarrow A(a) \wedge B(b)$. Επιπρόσθετα, ισχύει $\lambda(\mathcal{C}'_2) = \mathcal{C}_3$.

($\Leftarrow$) Αποδεικνύουμε το αντίστροφο. Έστω μία παραγωγή $\mathcal{C}'_1, \ldots, \mathcal{C}'_n$ από το $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$. Θα δείξουμε ότι υπάρχει μία παραγωγή $\mathcal{C}_1, \ldots, \mathcal{C}_m, n \leq m$ από το DD($\mathcal{S}(\mathcal{T} \cup$

$\{\mathcal{Q}\}) \cup \mathcal{A})$ τέτοια ώστε για κάθε $\mathcal{C}'_j$ υπάρχει μία πρόταση $\mathcal{C}_i$ με $\mathcal{C}_i = \lambda(\mathcal{C}'_j), j \leq i$. Αποδεικνύουμε με επαγωγή στο βάθος j της παραγωγής από το $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A}$.

Βασική υπόθεση: για $j = 0$ ο ισχυρισμός προκύπτει από τον ορισμό του DD.

Επαγωγικό βήμα: για $j = k$ υποθέτουμε ότι υπάρχει μία πρόταση $\mathcal{C}_i$ τέτοια ώστε $\mathcal{C}_i = \lambda(\mathcal{C}'_k), k \leq i$. Θα αποδείξουμε ότι ο ισχυρισμός ισχύει για $j = k + 1$. Θεωρούμε τους κανόνες $\mathcal{C}'_k, \mathcal{C}'_{sp} \vdash \mathcal{C}'_{k+1}$. Προκειμένου να κατασκευαστεί το $\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\})$ έχει ήδη εφαρμοστεί κάθε πιθανός κανόνας με δευτερεύουσα προκείμενη που δεν είναι βασική. Επομένως, η $\mathcal{C}'_{sp}$ είναι βασική πρόταση.

Αρχικά, θεωρούμε την περίπτωση κανόνα θετικής υπέρθεσης, έστω $\mathcal{C}'_k, \mathcal{C}'_{sp} \vdash \mathcal{C}'_{k+1}$. Αν $\mathcal{C}'_k$ είναι της μορφής $R(a, b) \leftarrow B(a), A(a)$ τότε η πρόταση $\mathcal{C}'_{sp}$ είναι $a \approx b$. Από την επαγωγική υπόθεση, οι προτάσεις $\mathcal{C}'_k, \mathcal{C}'_{sp}$ παράγονται επίσης από το DD$(\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A})$ και έτσι κάθε βασική πρόταση $\mathcal{C}'_{k+1}$ παράγεται από το DD$(\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A})$. Έστω η $\mathcal{C}'_k$ με τύπο $2.1 - 2.3$, της μορφής $R(x, f(x)) \leftarrow A(x)$ ή $A(f(x)) \leftarrow B(x)$, τότε η $\mathcal{C}'_{sp}$ είναι της μορφής $f(a) \approx b \leftarrow C(a) \wedge D(b)$. Από την επαγωγική υπόθεση η $\mathcal{C}_{sp}$, τέτοια ώστε $\lambda(\mathcal{C}'_{sp}) = \mathcal{C}_{sp}$, μπορεί να παραχθεί από το DD$(\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A})$. Από τον ορισμό του DD, ισχύει ότι υπάρχει πρόταση $\mathcal{C}_i$, τέτοια ώστε $\lambda(\mathcal{C}'_k) = \mathcal{C}_i$, και άρα, η $\mathcal{C}_i$ είναι της μορφής $R(x, x_f) \leftarrow A(x) \wedge S_f(x, x_f)$, ή $A(x_f) \leftarrow B(x) \wedge S(x, x_f)$, καθώς και η πρόταση $S_f(a, a_f)$. Η επίλυση στις $\mathcal{C}_i$, $S_f(a, a_f)$ παράγει την $\mathcal{C}_{i+1}$ της μορφής $R(a, a_f) \leftarrow A(a)$, ή την $A(a_f) \leftarrow \mathbf{B}(a) \wedge D(b)$. Στη συνέχεια, υπέρθεση με τις $\mathcal{C}_{i+1}, \mathcal{C}_{sp}$ παράγει την $\mathcal{C}_{i+2}$, με $\mathcal{C}_{i+2} = \mathcal{C}'_{k+1}$, και επομένως, ισχύει $\mathcal{C}_{i+2} = \lambda(\mathcal{C}'_{k+1})$.

Στη συνέχεια, εξετάζουμε την περίπτωση που η $\mathcal{C}'_k$ συμμετέχει σε εφαρμογή κανόνα επίλυσης, $\mathcal{C}'_k, \mathcal{C}'_{sp} \vdash \mathcal{C}'_{k+1}$. Με βάση την επαγωγική υπόθεση μία πρόταση $\mathcal{C}_i$, τέτοια ώστε $\lambda(\mathcal{C}'_k) = \mathcal{C}_i$, παράγεται από το DD$(\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\}) \cup \mathcal{A})$. Η $\mathcal{C}'_k$ δεν περιλαμβάνει συναρτήσεις στο σώμα της εφόσον επιλύεται με βασική πρόταση. Επιπρόσθετα, η $\mathcal{C}'_k$ δεν μπορεί να έχει κάποιο από τους τύπους $2.1 - 2.3, 6.1, 6.2$, καθώς όταν οι προτάσεις αυτού του τύπου δεν περιλαμβάνουν στο σώμα τους συναρτήσεις, έχουν μέγιστο άτομο στην κεφαλή. Επομένως, είναι φανερό ότι $\mathcal{C}_i = \mathcal{C}'_k$. Η δευτερεύουσα προκείμενη $\mathcal{C}'_{sp}$ είναι της μορφής $R(a, b) \leftarrow A(a) \wedge B(b)$, ή $A(a)$ και έτσι από την ΕΥ1 έχουμε ότι $\mathcal{C}_{sp} = \mathcal{C}'_{sp}$. Για το συμπέρασμα της επίλυσης της $\mathcal{C}_i$ με την $\mathcal{C}_{sp}$ ισχύει $\mathcal{C}_{i+i} = \mathcal{C}'_{k+1}$. Είναι φανερό ότι $\lambda(\mathcal{C}'_{k+1}) = \mathcal{C}_{i+1}$.

□

Θεώρημα 5.1.12. *Έστω $\mathcal{T}$ ένα Horn-$\mathcal{SHIQ}$ TBox, $\mathcal{A}$ ένα ABox, και $\mathcal{Q}$ ένα ΣΕ. Το* DD$(\mathcal{S}(\mathcal{T} \cup \{\mathcal{Q}\} \cup \mathcal{A})$ *είναι μία επαναγραφή του $\mathcal{Q}$ με βάση το $\mathcal{T}$.*

Παράδειγμα 5.1.13. Η datalog επαναγραφή DD$(\mathcal{S}(\mathcal{T}))$ του TBox $\mathcal{T}$ του Παραδείγματος 2.4.4 περιλαμβάνει την (2.8), και τις ακόλουθες προτάσεις:

$$
\begin{aligned}
(2.9) &\rightsquigarrow R(x, x_f) \leftarrow K(x) \wedge S_f(x, x_f) \\
(2.10) &\rightsquigarrow B(x_f) \leftarrow K(x) \wedge S_f(x, x_f) \\
(2.11) &\rightsquigarrow R(x_h, x) \leftarrow M(x) \wedge S_h(x, x_h) \\
(5.2) &\rightsquigarrow x_f \approx z \leftarrow K(x) \wedge R(x, z) \wedge B(z) \wedge S_f(x, x_f)
\end{aligned}
$$

◊

5.2 Βελτιστοποιημένος αλγόριθμος επαναγραφής για την Horn-$\mathcal{SHIQ}$, $\mathcal{I}_{\mathrm{RHS}}$

Όπως περιγράψαμε στην προηγούμενη ενότητα, η βασική ιδέα που διέπει τον αλγόριθμο $\mathcal{I}_{\mathrm{HS}}$, είναι η άμεση απομάκρυνση των συναρτησιακών συμβόλων που εμφανίζονται στις προτάσεις που παράγονται με τους κανόνες της επίλυσης. Όπως είδαμε στο Παράδειγμα 5.1.2, κάτι τέτοιο δεν είναι πάντοτε εφικτό. Στο επόμενο παράδειγμα παρουσιάζουμε μία ακόμα παραγωγή που κατασκευάζει ο $\mathcal{I}_{\mathrm{HS}}$.

Παράδειγμα 5.2.1. Έστω το σώμα ορολογίας $\mathcal{T}$ του Παραδείγματος 2.4.4. Σε αυτό το παράδειγμα, θεωρούμε την περίπτωση που η συνάρτηση επιλογής του $\mathcal{I}_{\mathrm{HS}}$ επιλέγει το άτομο $R(x, y)$ της 2.8 και εφαρμόζεται ο παρακάτω κανόνας επίλυσης:

$$(2.8), (2.11) \vdash y \approx z \leftarrow M(x) \wedge R(h(y), z) \wedge B(y) \wedge B(z) \tag{5.3}$$

Στη συνέχεια, επιλέγεται το άτομο $R(h(y), z)$ που περιλαμβάνει τον όρο με συνάρτηση και εφαρμόζεται ο ακόλουθος κανόνας επίλυσης:

$$(5.3), (2.9) \vdash f(h(y)) \approx y \leftarrow M(x) \wedge K(h(y)) \wedge B(y) \wedge B(f(h(y))) \tag{5.4}$$

Για του ίδιους λόγους, επιλέγεται το $B(f(h(y)))$:

$$(5.4), (2.10) \vdash f(h(y)) \approx y \leftarrow M(x) \wedge K(h(y)) \wedge B(y) \tag{5.5}$$

Εφόσον δεν είναι δυνατό να παραχθεί κάποια πρόταση της μορφής $K(h(x)) \leftarrow A(x)$ η παραγωγή τερματίζει. ♢

Παρατηρούμε ότι οι προτάσεις (5.3)–(5.5) δεν επιλύονται με κάποια βασική πρόταση, και επομένως δεν υπάρχει λόγος να συνυπολογιστούν στο αποτέλεσμα της επαναγραφής. Όπως γίνεται κατανοητό, η παραγωγή που περιγράψαμε δεν είναι χρήσιμη ως προς την επαναγραφή. Οι οντολογίες του πραγματικού κόσμου περιλαμβάνουν πολλές προτάσεις της μορφής $R(v, f(v)) \leftarrow K(v)$, που αλληλεπιδρούν με τις προτάσεις με ισότητα, γεγονός που μπορεί να οδηγήσει σε περιττά μονοπάτια παραγωγής, όπως συμβαίνει στο παραπάνω παράδειγμα. Ιδανικά, ένας λογισμός επαναγραφής παράγει προτάσεις με συναρτήσεις μόνο όταν αυτές είναι απαραίτητες, με την έννοια ότι συμβάλλουν στην παραγωγή άλλων προτάσεων που συμπεριλαμβάνονται στο σύνολο της επαναγραφής. Στην περίπτωση της $\mathcal{ELHI}$, όπως παρουσιάσαμε στην Ενότητα 4, κάτι τέτοιο επιτυγχάνεται με τον κανόνα της συρρίκνωσης. Ακουλουθώντας την ίδια λογική με αυτή που διέπει τον $\mathcal{I}_{\mathcal{EL}}$, σχεδιάζουμε έναν αλγόριθμο για τη Horn-$\mathcal{SHIQ}$, επεκτείνοντας τον κανόνα της συρρίκνωσης και της εκτύλιξης ώστε να εφαρμόζονται σε προτάσεις με ισότητα.

Παράδειγμα 5.2.2. Θεωρούμε και πάλι την οντολογία του Παραδείγματος 2.4.4. Ο κανόνας της συρρίκνωσης με κύρια προκείμενη την (2.8) και δευτερεύουσες τις (2.9), (2.10) παράγει την πρόταση (5.2). Επιπλέον, λόγω των συνθηκών του κανόνα της συρρίκνωσης, η πρόταση (5.2) δεν επιλύεται με την (2.11) σε αντίθεση με τους κανόνες που εφαρμόζονται από τον $\mathcal{I}_{\mathrm{HS}}$, όπως φαίνεται στο Παράδειγμα 5.1.2.

Όπως μπορούμε να παρατηρήσουμε, ο κανόνας της συρρίκνωσης δεν παράγει ποτέ προτάσεις με τύπο 6.3 της μορφής 5.3, ή προτάσεις με τύπο 6.2 της μορφής 5.4. Είναι φανερό, ότι εισάγοντας τον κανόνα της συρρίκνωσης στο λογισμό μας για τη Horn-$\mathcal{SHIQ}$ αποφεύγουμε τις παραγωγές που οδηγούν σε αδιέξοδο, καθώς και τις προτάσεις που περιλαμβάνουν όρους συνάρτησης με βάθος μεγαλύτερο από 1.

Παράδειγμα 5.2.3. Θεωρούμε μία Horn-$\mathcal{SHIQ}$ οντολογία που περιλαβάνει τις παρακάτω προτάσεις:

$$y \approx z \leftarrow A(x) \wedge R(x,y) \wedge R(x,z) \wedge B(y) \wedge B(z) \tag{5.6}$$

$$R(x, g(x)) \leftarrow D(x) \tag{5.7}$$

$$S(x, f(x)) \leftarrow C(x) \tag{5.8}$$

$$B(f(x)) \leftarrow C(x) \tag{5.9}$$

$$R(x,y) \leftarrow S(x,y) \tag{5.10}$$

Αρχικά, ο κανόνας της εκτύλιξης εφαρμόζεται στις (5.6), (5.10) με αποτέλεσμα την πρόταση (5.11). Ο κανόνας της συρρίκνωσης με δευτερεύουσες τις (5.8), (5.9) παράγει την (5.12).

$$(5.6), (5.10) \vdash \quad y \approx z \leftarrow A(x) \wedge S(x,y) \wedge R(x,z) \wedge B(y) \wedge B(z) \tag{5.11}$$

$$(5.11), (5.8), (5.9) \vdash \quad f(x) \approx z \leftarrow A(x) \wedge C(x) \wedge R(x,z) \wedge B(z) \tag{5.12}$$

Παρατηρούμε πως οι προτάσεις (5.6) και (5.7) δεν επιλύονται λόγω του κανόνα της συρρίκνωσης.

Στην περίπτωση της Horn-$\mathcal{SHIQ}$ η συλλογιστική περιλαμβάνει και τον κανόνα της θετικής υπέρθεσης λόγω των προτάσεων με ισότητα. Ο κανόνας της θετικής υπέρθεσης εφαρμόζεται με δευτερεύουσες προκείμενες που έχουν τύπο 6.1 και 6.2, της μορφής $f(x) \approx g(x) \leftarrow \mathbf{A}(x)$ και $f(g(x)) \approx x \leftarrow \mathbf{A}(x)$. Οι προτάσεις αυτές δεν περιλαμβάνουν άτομα που επιλέγονται από τη συνάρτηση παραγωγής, όπως ορίζεται στον $\mathcal{I}_{\mathrm{HS}}$, και διαθέτουν αυστηρά μέγιστη κεφαλή. Επιπλέον, οι κύριες προκείμενες έχουν τύπο 2.1–2.3 και περιλαμβάνονται στο αρχικό σώμα ορολογίας TBox, εφόσον τα συναρτησιακά σύμβολα που διαθέτουν δεν πρέπει με βάση τον ορισμό της βασικής υπέρθεσης, να έχουν εισαχθεί από κάποια αντικατάσταση. Προφανώς, δεν περιλαμβάνουν τα άτομα $\mathbf{B}(f(x))$ στο σώμα. Στη συνέχεια, διακρίνουμε τους διαφορετικούς τύπους κανόνων θετικής υπέρθεσης που εφαρμόζονται στον $\mathcal{I}_{\mathrm{HS}}$.

Η υπέρθεση των προτάσεων της μορφής $f(x) \approx g(x) \leftarrow \mathbf{A}(x)$, σε προτάσεις με τύπο 2.1–2.3, παράγει προτάσεις ίδιου τύπου που διαθέτουν συναρτήσεις μόνο στην κεφαλή. Η αλληλεπίδραση αυτή περιγράφεται στον κανόνα θετικής υπέρθεσης που ονομάζουμε κανόνα υπέρθεσης-1, και θα ορίσουμε στη συνέχεια (Ορισμός 5.2.4).

Αντίθετα, η υπέρθεση μίας πρότασης $f(g(x)) \approx x \leftarrow B(x)$ σε μία άλλη με τύπο 2.1, της μορφής $R(x, f(x)) \leftarrow A(x)$, παράγει την $R(g(x), x) \leftarrow A(g(x)) \wedge B(x)$ που περιέχει άτομο με συνάρτηση στο σώμα. Σύμφωνα με τον $\mathcal{I}_{\mathrm{HS}}$ επιλέγεται το άτομο $A(g(x))$ προκειμένου να απομακρυνθεί το σύμβολο συνάρτησης. Η πρόταση αυτή είναι χρήσιμη για την επαναγραφή μόνο στην περίπτωση που είναι δυνατή η

παραγωγή μίας πρότασης της μορφής $A(g(x)) \leftarrow D(x)$ από τον κανόνα της επίλυσης. Η εφαρμογή ενός κανόνα υπέρθεσης ακολουθούμενη από κανόνα επίλυσης μπορεί να συγχωνευτεί σε ένα βήμα που ονομάζουμε υπέρθεσης-2, και θα ορίσουμε στη συνέχεια (Ορισμός 5.2.4).

Ορισμός 5.2.4. Ο λογισμός $\mathcal{I}_{\mathrm{RH}\mathcal{S}}$ ορίζεται από το σύνολο των κανόνων του Πίνακα 5.3,[1] όπου κάθε πρόταση $\mathcal{C}_{(i)}$ μπορεί να έχει έναν από τους τύπους 2.1–2.3, 3.1–3.3, χωρίς να περιλαμβάνει συναρτήσεις στο σώμα. Επιπλέον, για ένα ΣΕ, $\mathcal{Q}$, ένα Horn-$\mathcal{SHIQ}$-TBox, $\mathcal{T}$, το Rapid-Hshiq$(\mathcal{Q}, \mathcal{T})$ υποδηλώνει το σύνολο που περιλαμβάνει τις προτάσεις με τύπο 2.1–2.3 στο $\mathcal{T}$, τις προτάσεις χωρίς συναρτήσεις, καθώς και όσες έχουν τύπο 6.3 και σώμα χωρίς συναρτήσεις και παράγονται από το $\mathcal{T} \cup \{\mathcal{Q}\}$ με τον $\mathcal{I}_{\mathrm{RH}\mathcal{S}}$. △

Παράδειγμα 5.2.5. Θεωρούμε το παρακάτω Horn-$\mathcal{SHIQ}$ TBox:

$$y \approx z \leftarrow A(x) \wedge R(x,y) \wedge R(x,z) \wedge B(y) \wedge B(z) \tag{5.13}$$

$$R(x, f(x)) \leftarrow A(x) \quad B(f(x)) \leftarrow A(x) \tag{5.14}$$

$$R(g(x), x) \leftarrow D(x) \quad A_1(g(x)) \leftarrow D(x) \tag{5.15}$$

$$A(x) \leftarrow A_1(x) \tag{5.16}$$

Σύμφωνα με τον $\mathcal{I}_{\mathrm{RH}\mathcal{S}}$ ο παρακάτω κανόνα εκτύλιξης εφαρμόζεται στην (5.13):

$$(5.13), (5.16) \quad \rightsquigarrow \quad y \approx z \leftarrow A_1(x) \wedge R(x,y) \wedge R(x,z) \wedge B(y) \wedge B(z) \tag{5.17}$$

Έπειτα, εφαρμόζεται ο παρακάτω κανόνας συρρίκνωσης:

$$(5.17), (5.14)(a), (5.14)(b) \quad \rightsquigarrow \quad f(x) \approx z \leftarrow A_1(x) \wedge R(x,z) \wedge B(z) \tag{5.18}$$

$$(5.18), (5.15)(a), (5.15)(b) \quad \rightsquigarrow \quad f(g(x)) \approx x \leftarrow D(x) \wedge B(x) \tag{5.19}$$

Η υπέρθεση με βάση τον $\mathcal{I}_{\mathrm{H}\mathcal{S}}$, της (5.19), στην (5.14)(a), ή την (5.14)(b) παράγει την $R(g(x), x) \leftarrow D(x) \wedge A(g(x))$, ή την $A_1(x) \leftarrow D(x) \wedge A(g(x))$. Επιπλέον, ο κανόνας της επίλυσης με τις (5.16), (5.15)(b) παράγει την $A(g(x)) \leftarrow D(x)$, η οποία επιλύεται με την $R(g(x), x) \leftarrow D(x) \wedge A(g(x))$ με αποτέλεσμα την πρόταση $R(g(x), x) \leftarrow D(x)$. Σύμφωνα με τον $\mathcal{I}_{\mathrm{RH}\mathcal{S}}$ ο κανόνας της θετικής υπέρθεσης συνδυάζεται με τους κανόνες της επίλυσης που εφαρμόστηκαν στον παρακάτω κανόνα υπέρθεσης *sup-2*:

$$(5.14)(a), (5.19), (5.16) \quad \rightsquigarrow \quad R(g(x), x) \leftarrow D(x) \tag{5.20}$$

◇

[1] Οι ρόλοι στον πίνακα μπορούν να αντικατασταθούν από τους ανάστροφους.

Πίνακας 5.3: Οι κανόνες του λογισμού $\mathcal{I}_{\mathrm{RHS}}$

εκτύλιξη (*unfolding*): $$\frac{\Upsilon \quad \mathcal{C}}{\Upsilon'\sigma}$$

όπου 1.Υ είναι είτε ένα ΣΕ, μία RA-πρόταση ή μία πρόταση με τύπο 5, $\Upsilon'\sigma$ είναι το συμπέρασμα των προτάσεων Υ και $\mathcal{C}$ που δεν περιλαμβάνει συναρτήσεις, και 2. αν $x \mapsto f(y) \in \sigma$, τότε $x \in \mathsf{unbound}(\Upsilon)$.

συρρίκνωση (*shrinking*): $$\frac{\Upsilon \quad \mathcal{C}_1\ [\mathcal{C}_2\ \ldots\ \mathcal{C}_n]}{\Upsilon'\sigma}$$

όπου 1. η Υ είναι ένα ΣΕ CQ, ή μία RA-πρόταση και το $\Upsilon'\sigma$ είναι το συμπέρασμα, χωρίς συναρτήσεις των $\Upsilon, \mathcal{C}_1, \ldots, \mathcal{C}_n$, ή η Υ είναι μία πρόταση με τύπο 5 ή 6.3 χωρίς συναρτήσεις στο σώμα και το $\Upsilon'\sigma$ είναι το συμπέρασμα των $\Upsilon, \mathcal{C}_1, \ldots, \mathcal{C}_n$, που δεν περιλαμβάνει συναρτήσεις στο σώμα, και 2. υπάρχει κάποια αντιστοίχιση $x \mapsto f(y) \in \sigma$ τ.ω. $x \notin \mathsf{unbound}(\Upsilon)$.

κανόνας-συνάρτησης (*function-rule*): $$\frac{B(x) \leftarrow R(x,y) \wedge [\mathbf{C}(y)] \quad R(f(x),x) \leftarrow A(x)}{B(f(x)) \leftarrow A(x) \wedge [\mathbf{C}(x)]}$$

υπέρθεση-1 (*sup-1:*) $$\frac{R(x,f(x)) \leftarrow B(x) \quad f(x) \approx g(x) \leftarrow \mathbf{D}(x)}{R(x,g(x)) \leftarrow B(x) \wedge \mathbf{D}(x)}$$ $$\frac{A(f(x)) \leftarrow B(x) \quad f(x) \approx g(x) \leftarrow \mathbf{D}(x)}{A(g(x)) \leftarrow B(x) \wedge \mathbf{D}(x)}$$

όπου η $f(x)$ δεν έχει εισαχθεί στην πρόταση από κάποια αντικατάσταση σε προηγούμενο βήμα.

υπέρθεση-2 (*sup-2*): $$\frac{R(x,f(x)) \leftarrow A_1(x) \quad \mathcal{G}_1\ \mathbf{L}\ \mathcal{G}_2}{R(g(x),x) \leftarrow \mathbf{C}(x) \wedge \mathbf{B}(x)}$$ $$\frac{B(f(x)) \leftarrow A_1(x) \quad \mathcal{G}_1\ \mathbf{L}\ \mathcal{G}_2}{B(x) \leftarrow \mathbf{C}(x) \wedge \mathbf{B}(x)}$$

όπου 1. $\mathcal{G}_1 = f(g(x)) \approx x \leftarrow \mathbf{B}(x)$, $\mathbf{L}$ είναι μία ενδεχομένως κενή, ακολουθία από m προτάσεις της μορφής $A_i(x) \leftarrow A_{i+1}(x)$, $\mathcal{G}_2 = A_i(g(x)) \leftarrow \mathbf{C}(x)$, και 2. η $f(x)$ δεν έχει εισαχθεί στην πρόταση από αντικατάσταση σε προηγούμενο βήμα.

Αλγόριθμος 3 Αλγόριθμος παραγωγής της επαναγραφής Rapid-Hshiq$(\mathcal{Q}, \mathcal{T})$

Είσοδος: ένα ΣΕ, $\mathcal{Q}$ και ένα TBox, $\mathcal{T}$ που περιλαμβάνει προτάσεις.

1: $\mathcal{T}_{el} := \{\rho \mid \rho \in \mathcal{T}$ τ.ω. η ρ είναι της μορφής $A(u) \leftarrow R(u,v) \wedge B(v)\}$
2: $\mathcal{T}_{eq} := \{\varepsilon \mid \varepsilon \in \mathcal{T}$ τ.ω. η ε είναι της μορφής $t_1 \approx t_2 \leftarrow \bigwedge D_i(\vec{s_i})\}$
3: $\mathcal{T}_{lt} := \mathcal{T} \setminus \{\mathcal{T}_{el} \cup \mathcal{T}_{eq}\}$
4: **repeat**
5: Επίλεξε μία πρόταση $\rho \in \mathcal{T}_{el}$
6: $\mathcal{P} :=$ REL$(\rho, \mathcal{T}_{lt})$ ▷ εκτύλιξη, συρρίκνωση, κανόνας συνάρτησης
7: $\mathcal{T}_{el} := \mathcal{T}_{el} \cup \{\rho \mid \rho \in \mathcal{P}$ τ.ω. η ρ να είναι της μορφής $A(u) \leftarrow R(u,v) \wedge B(v)\}$
8: $\mathcal{T}_{lt} := \mathcal{T}_{lt} \cup (\mathcal{P} \setminus \mathcal{T}_{el})$
9: Επίλεξε μία πρόταση $\varepsilon \in \mathcal{T}_{eq}$
10: $\mathcal{T}_{eq} :=$ RLite$(\varepsilon, \mathcal{T}_{lt})$ ▷ εκτύλιξη, συρρίκνωση
11: $\mathcal{E} :=$ RHshiq$(c, \mathcal{T}_{eq} \cup \mathcal{T}_{lt}), \forall c \in \mathcal{T}$ της μορφής $R(v, f(v)) \leftarrow A(v), C(f(v)) \leftarrow A(v)$ ▷ υπέρθεση-1, υπέρθεση-2
12: $\mathcal{T}_{lt} := \mathcal{T}_{lt} \cup (\mathcal{E} \setminus \mathcal{T}_{eq})$
13: **until** λαμβάνοντας υπόψη τις μετονομασίες των μεταβλητών, δεν προστίθενται νέες προτάσεις στα $\mathcal{T}_{el}$, $\mathcal{T}_{lt}$ και $\mathcal{T}_{eq}$.
14: $\mathcal{R}_Q := \{\mathcal{Q}\}$
15: **repeat**
16: Επίλεξε μία πρόταση $\mathcal{Q}' \in \mathcal{R}_Q$
17: $\mathcal{R}_Q := \mathcal{R}_Q \cup$ RLite$(\mathcal{Q}', \mathcal{T}_{lt})$
18: **until** δεν προστίθεται κανένα νέο ερώτημα στο $\mathcal{R}_Q$ λαμβάνοντας υπόψη και τις μετονομασίες των μεταβλητών
19: **return** $\mathcal{R}_Q \cup \mathcal{T}_{el} \cup$ DD$(\mathcal{T}_{lt} \cup \mathcal{T}_{eq})$

Από τα παραπάνω, παρατηρούμε ότι στην περίπτωση της Horn-$\mathcal{SHIQ}$, όπως και στην $\mathcal{ELHI}$, και σε αντίθεση με ότι συμβαίνει στην DL-Lite, ο αλγόριθμος επαναγραφής $\mathcal{I}_{\mathrm{RH}\mathcal{S}}$, εφαρμόζει τους κανόνες της επίλυσης αρχικά σε ένα σύνολο προτάσεων του TBox, και συγκεκριμένα, στις RA-προτάσεις και τις προτάσεις με ισότητα. Έπειτα σε δεύτερο στάδιο, παράγονται τα ερωτήματα που θα συμπεριληφθούν στην έξοδο του αλγορίθμου. Πιο αναλυτικά, στη γραμμή 10, εφαρμόζονται οι κανόνες του $\mathcal{I}_{lite}$ στις προτάσεις με ισότητα (διαδικασία RLite$(\varepsilon, \mathcal{T}_{lt})$). Στη συνέχεια, γραμμή 11, οι προτάσεις που προέκυψαν χρησιμοποιούνται από τους κανόνες υπέρθεσης (RHshiq$(c, \mathcal{T}_{eq} \cup \mathcal{T}_{lt})$). Όπως και στην $\mathcal{ELHI}$, οι κανόνες του $\mathcal{I}_{lite}$ εφαρμόζονται στις RA-προτάσεις (γραμμή 7). Το σύνολο $\mathcal{T}_{lt}$ ενημερώνεται με τις προτάσεις που προκύπτουν από την αρχική φάση του αλγορίθμου που αφορά μόνο προτάσεις του TBox, ώστε να ακολουθήσει η παραγωγή των ερωτημάτων της επαναγραφής (γραμμή 17).

5.3 Ορθότητα του $\mathcal{I}_{\mathrm{RH}\mathcal{S}}$

Η ορθότητα του αλγορίθμου $\mathcal{I}_{\mathrm{RH}\mathcal{S}}$ προκύπτει μετασχηματίζοντας τις παραγωγές που κατασκευάζονται από τον $\mathcal{I}_{\mathrm{H}\mathcal{S}}$, ο οποίος γνωρίζουμε ότι είναι πλήρης και ορθός λογισμός, σε παραγωγές από τον $\mathcal{I}_{\mathrm{RH}\mathcal{S}}$. Ο μετασχηματισμός αυτός πραγματοποιείται σε δύο βήματα. Αρχικά, από τη μελέτη του Ορισμού 5.2.4, όπως αναφέρθηκε και νωρίτερα, μπορούμε να συμπεράνουμε ότι ο λογισμός $\mathcal{I}_{\mathrm{RH}\mathcal{S}}$ εφαρμόζει μία τεχνική SLD-επίλυσης, χρησιμοποιώντας δευτερεύουσες προκείμενες από τη δεδομένη οντολογία εισόδου, με ορισμένες εξαιρέσεις. Συγκεκριμένα, οι εξαιρέσεις αφορούν τις προτάσεις που παράγονται από τους κανόνες υπέρθεσης-1 και υπέρθεσης-2, και τον κανόνα-συνάρτησης. Οι κύριες προκείμενες είναι προτάσεις με τύπο 1, 5, ή 6.3 χωρίς συναρτήσεις στο σώμα τους, ή RA-προτάσεις. Παρουσιάζουμε τυπικά αυτές τις παραγωγές στον παρακάτω Ορισμό 5.3.1 ο οποίος επεκτείνει τον Ορισμό 4.2.1.

Ορισμός 5.3.1. Έστω Σ ένα σύνολο από Horn προτάσεις. Μία *επεκταμένη*-SLD παραγωγή από το Σ είναι μία ακολουθία προτάσεων $\mathcal{C}_1, \ldots, \mathcal{C}_n$ τέτοια ώστε κάθε πρόταση $\mathcal{C}_i$ αντιστοιχεί σε μία από τις παρακάτω περιπτώσεις:

- έχει τύπο 1 (ΣE), ή είναι μία RA-πρόταση, ή έχει τύπο 5 και ανήκει στο Σ.
- είναι το συμπέρασμα ενός κανόνα δυαδικής επίλυσης με κύρια προκείμενη με τύπο 1, ή τύπο 5, ή τύπο 6.3, ή μία RA-πρόταση, ή προτάσεις με τύπο 2.1-2.2, 3.3, 6.1-6.2 με συνάρτηση στο σώμα από το σύνολο $\{\mathcal{C}_1, \ldots, \mathcal{C}_{i-1}\}$, και ως δευτερεύουσα προκείμενη μία πρόταση από το Σ, ή μία πρόταση χωρίς συναρτήσεις με τύπο 3.3 από το $\{\mathcal{C}_1, \ldots, \mathcal{C}_{i-1}\}$, ή μία πρόταση που παράγεται από τον κανόνα συνάρτησης με δευτερεύουσα προκείμενη από το $\{\mathcal{C}_1, \ldots, \mathcal{C}_{i-1}\}$, ή μια πρόταση με τύπο 2.1, 2.2, ή 2.3 χωρίς συναρτήσεις στο σώμα από το $\{\mathcal{C}_1, \ldots, \mathcal{C}_{i-1}\}$.
- είναι το συμπέρασμα ενός κανόνα βασικής υπέρθεσης με κύρια προκείμενη μία πρόταση με τύπο 2.1, 2.2, ή 2.3 από το Σ και σα δευτερεύουσα προκείμενη μία πρόταση με τύπο 6.1, ή 6.2 με χώμα χωρίς συναρτήσεις από το $\{\mathcal{C}_1, \ldots, \mathcal{C}_{i-1}\}$.

Συμβολίζουμε το σύστημα που κατασκευάζει τέτοιου είδους παραγωγές με $\mathcal{I}_{\mathsf{ESLD}}$. $\triangle$

Λήμμα 5.3.2. *Έστω ένα Horn-$\mathcal{SHIQ}$ TBox $\mathcal{T}$. Θεωρούμε ένα κανόνα υπέρθεσης $\mathcal{C}_1, \Upsilon \vdash^{\mathcal{I}_{\mathrm{H}\mathcal{S}}} \mathcal{C}$. Κάθε πρόταση $\mathcal{C}'$ που δε διαθέτει συναρτήσεις στο σώμα της, που παράγεται από το $\mathcal{T}$ με τον $\mathcal{I}_{\mathrm{H}\mathcal{S}}$ ξεκινώντας από την $\mathcal{C}$, παράγεται επίσης με τον $\mathcal{I}_{\mathsf{ESLD}}$.*

Απόδειξη. Από τον ορισμό του $\mathcal{I}_{\mathrm{H}\mathcal{S}}$, η πρόταση $\mathcal{C}_1$ έχει τύπο 2.1, 2.2, ή 2.3 και ανήκει στο $\mathcal{T}$, και η πρόταση $\mathcal{K}$ με τύπο 6.1, ή 6.2 δε διαθέτει συναρτήσεις στο σώμα. Επίσης, η πρόταση $\mathcal{C}$ έχει τύπο 2.1, ή 2.2 και είναι της μορφής $R(x, f(x)) \leftarrow A(x) \wedge [B(f(x))]$, ή τύπο 2.3 και είναι της μορφής $A(f(x)) \leftarrow B(x)$, ή τύπο 3.3 της μορφής $A(x) \leftarrow \wedge B(x) \wedge [C(f(x))]$. Στην περίπτωση που η $\mathcal{K}$ έχει τύπο 6.1, η $\mathcal{C}$ δεν έχει συναρτήσεις στο σώμα, και η $\mathcal{C}'$ συμπίπτει με την $\mathcal{C}$. Αποδεικνύουμε την πρόταση επαγωγικά. Υποθέτουμε ότι σε κάποιο βάθος της παραγωγής η $\mathcal{C}'$ που δε διαθέτει συναρτήσεις στο σώμα της, και παράγεται από το $\mathcal{T}$ με τον $\mathcal{I}_{\mathrm{H}\mathcal{S}}$ ξεκινώντας από μία πρόταση $\mathcal{C}$ που παράγεται από κανόνα υπέρθεσης $\mathcal{C}_1, \Upsilon \vdash \mathcal{C}$, είναι δυνατό να παραχθεί από το $\mathcal{T}$ μέσω

$\mathcal{I}_{\mathsf{ESLD}}$ (IH1). Θεωρούμε μία νέα πρόταση $\mathcal{C}'$ που παράγεται με τον $\mathcal{I}_{\mathrm{H}\mathcal{S}}$, ξεκινώντας απο ένα κανόνα υπέρθεσης $\mathcal{C}_1, \Upsilon \vdash \mathcal{C}$. Σύμφωνα με τον Πίνακα 5.2, οι προτάσεις με τύπο 6.1, 6.2 είναι δυνατό να προκύψουν μόνο από μία πρόταση 6.3, ενώ μία πρόταση με τύπο 6.3 είναι δυνατό να προκύψει μόνο από κάποια με τύπο 5. Αρχικά, αποδεικνύουμε τους παρακάτω ισχυρισμούς:

> *Ισχυρισμός 1*: Κάθε πρόταση με τύπο 3.3 που παράγεται από το $\mathcal{T}$ με βάση το λογισμό $\mathcal{I}_{\mathrm{H}\mathcal{S}}$, ξεκινώντας από μία RA-πρόταση, $\mathcal{A}$, παράγεται επίσης, από το $\mathcal{T}$ με τον $\mathcal{I}_{\mathsf{ESLD}}$ ξεκινώντας από την $\mathcal{A}$.
>
> *Ισχυρισμός 2*: Κάθε πρόταση $\mathcal{F}$ με τύπο 6.3, που παράγεται από το $\mathcal{T}$ μέσω του $\mathcal{I}_{\mathrm{H}\mathcal{S}}$, ξεκινώντας με μία πρόταση $\mathcal{L}$ με τύπο 5 παράγεται από το $\mathcal{T}$ μέσω του $\mathcal{I}_{\mathsf{ESLD}}$ ξεκινώντας από την $\mathcal{L}$.
>
> *Ισχυρισμός 3*: Κάθε πρόταση Υ με τύπο 6.1 ή 6.2 που παράγεται από το $\mathcal{T}$ μέσω του $\mathcal{I}_{\mathrm{H}\mathcal{S}}$ ξεκινώντας από την $\mathcal{F}$ με τύπο 6.3, παράγεται επίσης από το $\mathcal{T}$ μέσω του $\mathcal{I}_{\mathsf{ESLD}}$, ξεκινώντας από την $\mathcal{F}$.

Απόδειξη του Ισχυρισμού 1: Αποδεικνύουμε επαγωγικά τον ισχυρισμό. Υποθέτουμε ότι σε κάποιο σημείο κάθε πρόταση με τύπο 3.3 που παράγεται από το $\mathcal{T}$ με τον $\mathcal{I}_{\mathrm{H}\mathcal{S}}$ ξεκινώντας από μία RA-πρόταση, παράγεται επίσης από το $\mathcal{T}$ με τον $\mathcal{I}_{\mathsf{ESLD}}$ (EΥ2). Υποθέτουμε ότι μία νέα πρόταση 3.3, έστω $\mathcal{E}$, παράγεται σε κάποιο σημείο της παραγωγής που ξεκινάει με την $\mathcal{A}$. Δείχνουμε μόνο την περίπτωση που η $\mathcal{A}$ είναι της μορφής $A(x) \leftarrow R(x,y) \wedge \mathbf{D}(y)$, δηλαδή με τύπο 4.1. Η απόδειξη για μία πρόταση $\mathcal{A}$ με τύπο 4.2 είναι όμοια.

Σύμφωνα με τον Πίνακα 5.2 η πρόταση $\mathcal{E}$ με τύπο 3.3 μπορεί να προκύψει από κανόνες της μορφής $\mathcal{E}_1, \mathcal{D}_1 \vdash^{\mathcal{I}_{\mathrm{H}\mathcal{S}}} \mathcal{E}$ όπου οι $\mathcal{E}_1$ και $\mathcal{D}_1$ έχουν κάποια μορφή από τις παρακάτω:

1. Η πρόταση $\mathcal{E}_1$ έχει τύπο 4.1 και η $\mathcal{D}_1$ 2.1; Η περίπτωση που η $\mathcal{E}_1$ έχει τύπο 4.2 και η $\mathcal{D}_1$ έχει τύπο 2.2 είναι όμοια. Εφόσον ο $\mathcal{I}_{\mathrm{H}\mathcal{S}}$ δεν παράγει ποτέ προτάσεις με τύπο 4.1, ισχύει ότι $\mathcal{E}_1 \in \mathcal{T}$. Αντιθέτως, η $\mathcal{D}_1$ παράγεται είτε από κανόνα υπέρθεσης και έτσι από την EΥ1 μπορεί να κατασκευαστεί από το $\mathcal{I}_{\mathsf{ESLD}}$, είτε από κανόνα της μορφής $\mathcal{J}_1, \mathcal{D}_2 \vdash \mathcal{D}_1$, όπου είτε η $\mathcal{J}_1$ έχει τύπο 3.1 και η $\mathcal{D}_2$ έχει και πάλι τύπο 2.1, ή η $\mathcal{J}_1$ έχει τύπο 2.2, της μορφής $R(f(x),x) \leftarrow A(x) \wedge B(f(x))$ και η $\mathcal{D}_2$ έχει τύπο 2.3. Στην τελευταία περίπτωση, η $\mathcal{J}_1$ κατασκευάζεται με υπέρθεση πρότασης με τύπο 6.2 σε πρόταση με τύπο 2.1, και από την EΥ1 η $\mathcal{D}_1$ μπορεί να προκύψει από το $\mathcal{I}_{\mathsf{ESLD}}$. Συνεπώς, εξετάζουμε την πρώτη περίπτωση, δηλαδή, την περίπτωση που η $\mathcal{J}_1$ έχει τύπο 3.1. Ο κανόνας $\mathcal{E}_1, \mathcal{D}_1 \vdash^{\mathcal{I}_{\mathrm{H}\mathcal{S}}} \mathcal{E}$ μπορεί να ξεδιπλωθεί στους κανόνες $\mathcal{E}_1, \mathcal{J}_1 \vdash \mathcal{E}_2$ και $\mathcal{E}_2, \mathcal{D}_2 \vdash \mathcal{E}$. Οι προτάσεις με τύπο 3.1 δεν παράγονται ποτέ από το $\mathcal{I}_{\mathrm{H}\mathcal{S}}$, και άρα $\mathcal{J}_1 \in \mathcal{T}$, ενώ η $\mathcal{D}_2$ (και πάλι με τύπο 2.1) παράγεται σε μικρότερο βάθος. Από ένα άμεσο επαγωγικό ισχυρισμό μπορούμε να δείξουμε ότι αυτοί οι κανόνες μπορούν να εκτυλιχθούν εξαντλητικά μέχρι να προκύψει κάποια παραγωγή $\mathcal{E}_1, \ldots, \mathcal{E}_n, \mathcal{E}$ όπου $\mathcal{E}_i, \mathcal{J}_i \vdash \mathcal{E}_{i+1}$, και οι προτάσεις $\mathcal{E}_i$ έχουν τύπο 4.1, με $\mathcal{E}_i \in \mathcal{T}$, όλες οι $\mathcal{J}_i$ έχουν τύπο 3.1 (ή 3.2) και ανήκουν επίσης στο $\mathcal{T}$, η $\mathcal{E}$ παράγεται από κάποιον κανόνα

$\mathcal{E}_n, \mathcal{D}_n \vdash \mathcal{E}$, όπου η $\mathcal{D}_n$ έχει τύπο 2.1 (ή 2.2) και πάλι ανήκει στο $\mathcal{T}$, ή παράγεται από υπέρθεση και άρα είναι δυνατό να προκύψει και με τον $\mathcal{I}_{\mathsf{ESLD}}$, σύμφωνα με την ΕΥ1. Συνεπώς, η $\mathcal{E}$ προκύπτει από το $\mathcal{T}$ με τον $\mathcal{I}_{\mathsf{ESLD}}$.

2. Η πρόταση $\mathcal{E}_1$ έχει τύπο 3.3, και είναι της μορφής $A(x) \leftarrow B(x) \wedge C(f(x))$, και η $\mathcal{D}_1$ έχει τύπο 2.3, της μορφής $C(f(x)) \leftarrow D(x)$. Εφόσον η $\mathcal{E}$ παράγεται από μία ακολουθία που ξενικά με την $\mathcal{A}$, το ίδιο ισχύει και για την $\mathcal{E}_1$ και έτσι από την ΕΥ2, η $\mathcal{E}_1$ παράγεται επίσης από το $\mathcal{T}$ με τον $\mathcal{I}_{\mathsf{ESLD}}$. Στρέφουμε τώρα την προσοχή μας στην $\mathcal{D}_1$. Από μία δεύτερη επαγωγή μπορούμε να δείξουμε ότι αν η $\mathcal{D}_1$ παράγεται από κανόνες σε προτάσεις με χαμηλότερο βάθος, τότε, είτε ο $\mathcal{E}_1, \mathcal{D}_1 \vdash^{\mathcal{I}_{\mathrm{HS}}} \mathcal{E}$ μπορεί να ξεδιπλωθεί χρησιμοποιώντας τις προτάσεις που παράγονται σε χαμηλότερο βάθος, είτε η $\mathcal{D}_1$ παράγεται με τον $\mathcal{I}_{\mathsf{ESLD}}$.

   Η $\mathcal{D}_1$ παράγεται από κανόνες της μορφής $\mathcal{K}, \mathcal{G} \vdash^{\mathcal{I}_{\mathrm{HS}}} \mathcal{D}_1$, όπου η $\mathcal{K}$ και η $\mathcal{G}$ έχουν κάποια από τις παρκάτω μορφές:

   (α′) η $\mathcal{K}$ έχει τύπο 2.3, ανήκει στο $\mathcal{T}$, και η $\mathcal{G}$ έχει τύπο 6.1, και άρα η $\mathcal{D}_1$ παράγεται με υπέρθεση και από την ΕΥ1, μπορεί να κατασκευαστεί από το $\mathcal{I}_{\mathsf{ESLD}}$.

   (β′) η $\mathcal{K}$ έχει τύπο 4.1 (4.2) και η $\mathcal{G}$ 2.2 (2.1). Δείχνουμε μόνο την περίπτωση του κανόνα 4.1+2.2. Η $\mathcal{K}$ είναι της μορφής $A(x) \leftarrow R(x, y) \wedge [\mathbf{C}(y)]$. Αρχικά, θεωρούμε ότι η $\mathcal{K}$ περιλαμβάνει ένα άτομο $C(y)$. Τότε, ο κανόνας $\mathcal{K}, \mathcal{G} \vdash^{\mathcal{I}_{\mathrm{HS}}} \mathcal{D}_1$ αντιστοιχεί στον κανόνα συνάρτησης και επιπλέον, εφόσον η $\mathcal{K}$ έχει τύπο 4.1, ισχύει ότι $\mathcal{K} \in \mathcal{T}$. Συνεπώς, αν επιπλέον η $\mathcal{G}$ παράγεται με υπέρθεση, και άρα παράγεται από το $\mathcal{I}_{\mathsf{ESLD}}$ με βάση την ΕΥ1, ή $\mathcal{G} \in \mathcal{T}$, τότε προκύπτει ότι η $\mathcal{D}_1$ παράγεται με τον $\mathcal{I}_{\mathsf{ESLD}}$. Επιπλέον, η $\mathcal{G}$ είναι δυνατό να παραχθεί από κανόνα που εφαρμόζεται στην πρόταση $\mathcal{J}_1$, με τύπο 2.2(2.1), της μορφής $R(f(x), x) \leftarrow A(x) \wedge B(f(x))$ και σε πρόταση με τύπο 2.3, $B(f(x)) \leftarrow C(x)$. Σε αυτή την περίπτωση, η $\mathcal{J}_1$ προκύπτει από κανόνα υπέρθεσης σε προτάσεις με τύπους 2.1, 6.2 και έτσι, από την ΕΥ1 έχουμε ότι η $\mathcal{G}$ παράγεται με τον $\mathcal{I}_{\mathsf{ESLD}}$. Αντιθέτως, η $\mathcal{G}$ μπορεί να παραχθεί από ένα κανόνα που εφαρμόζεται στις προτάσεις $\mathcal{J}_1$, με τύπο 3.1(3.2), και $\mathcal{G}_1$ με τύπο 2.2(2.1). Τότε, ο κανόνας $\mathcal{K}, \mathcal{G} \vdash \mathcal{D}_1$ εκτυλίσσεται στους $\mathcal{K}, \mathcal{J}_1 \vdash \mathcal{K}_1$ και $\mathcal{K}_1, \mathcal{G}_1 \vdash \mathcal{D}_1$. Επιπλέον, ισχύει ότι $\mathcal{J}_1 \in \mathcal{T}$, αφού οι προτάσεις αυτές δεν κατασκευάζονται από το $\mathcal{I}_{\mathrm{HS}}$, και έτσι, η $\mathcal{K}_1$ παράγεται με τον $\mathcal{I}_{\mathsf{SLD}}$ από την $\mathcal{K}$. Μπορούμε να εφαρμόσουμε την ίδια διαδικασία εκτύλιξης εξαντλητικά μέχρι να προκύψει μία παραγωγή της μορφής $\mathcal{K}_0, \mathcal{K}_1, \ldots, \mathcal{K}_n, \mathcal{D}_1$, όπου για $0 \leq i < n$ έχουμε ότι $\mathcal{K}_i, \mathcal{J}_{i+1} \vdash \mathcal{K}_{i+1}$, $\mathcal{K}_0 = \mathcal{K}$, οι $\mathcal{J}_{i+1}$ έχουν τύπο 3.1, και $\mathcal{K}_n, \mathcal{G}_n \vdash \mathcal{D}_1$ για κάποια $\mathcal{G}_n$ που περιλαμβάνεται στο $\mathcal{T}$, ή έχει προκύψει από υπέρθεση και άρα από την ΕΥ1 έχουμε ότι είναι δυνατό να παραχθεί με τον $\mathcal{I}_{\mathsf{ESLD}}$. Εφόσον όλες οι $\mathcal{J}_i$ έχουν τύπο 3.1 και $\mathcal{K}_0 \in \mathcal{T}$, ισχύει ότι η $\mathcal{K}_n$ παράγεται με τον $\mathcal{I}_{\mathsf{ESLD}}$.

   Στην περίπτωση που η $\mathcal{K}$ είναι της μορφής $A(x) \leftarrow R(x, y)$, τότε ο κανόνας $\mathcal{K}, \mathcal{G} \vdash \mathcal{D}_1$ μπορεί να χρησιμοποιηθεί προκειμένου να εκτυλιχθεί ο κανόνας $\mathcal{E}_1, \mathcal{D}_1 \vdash \mathcal{E}$ στους $\mathcal{E}_1, \mathcal{K} \vdash \mathcal{E}'$, $\mathcal{E}', \mathcal{G} \vdash \mathcal{E}$ όπου οι $\mathcal{G}, \mathcal{K}$ παράγονται σε

μικρότερα βάθη.

(γ′) Η $\mathcal{K}$ έχει τύπο 3.3, και είναι της μορφής $C(x) \leftarrow \mathbf{B}(x)$ και η $\mathcal{G}$ έχει τύπο 2.3, της μορφής $B(f(x)) \leftarrow D(x)$. Σε αυτή την περίπτωση ο $\mathcal{E}_1, \mathcal{D}_1 \vdash \mathcal{E}$ εκτυλίσσεται στους $\mathcal{E}_1, \mathcal{K} \vdash \mathcal{E}'$ και $\mathcal{E}', \mathcal{G} \vdash \mathcal{E}$, και οι $\mathcal{K}$, $\mathcal{G}$ παράγονται σε χαμηλότερα βάθη.

(δ′) οι $\mathcal{K}$ και $\mathcal{G}$ έχουν τύπο 2.3. Τότε ο $\mathcal{E}_1, \mathcal{D}_1 \vdash \mathcal{E}$ εκτυλίσσεται στους $\mathcal{E}_1, \mathcal{K} \vdash \mathcal{E}'$, $\mathcal{E}', \mathcal{G} \vdash \mathcal{E}$ όπου οι $\mathcal{K}$ and $\mathcal{G}$ παράγονται σε χαμηλότερα βάθη.

3. Η πρόταση $\mathcal{E}_1$ έχει τύπο 3.3 και είναι της μορφής $A(x) \leftarrow B(x) \wedge C(f(x))$ και η πρόταση $\mathcal{D}_1$ έχει τύπο 3.3, της μορφής $C(x) \leftarrow \mathbf{D}(x)$. Εφόσον η $\mathcal{E}$ παράγεται από μία ακολουθία που ξεκινάει με την $\mathcal{A}$, το ίδιο συμβαίνει και με την $\mathcal{E}_1$, και έτσι, από την ΕΥ2, η $\mathcal{E}_1$ παράγεται από το $\mathcal{T}$ με τον $\mathcal{I}_{\mathsf{ESLD}}$. Στρέφουμε τώρα το ενδιαφέρον μας στην $\mathcal{D}_1$. Από μία δεύτερη επαγωγή μπορούμε να δείξουμε ότι η $\mathcal{D}_1$ παράγεται από κανόνες σε άλλες προτάσεις, τότε ο $\mathcal{E}_1, \mathcal{D}_1 \vdash^{\mathcal{I}_{\mathrm{HS}}} \mathcal{E}$ είναι δυνατό είτε να εκτυλιχθεί χρησιμοποιώντες αυτές τις προτάσεις, είτε η $\mathcal{D}_1$ παράγεται από το $\mathcal{I}_{\mathsf{ESLD}}$. Η $\mathcal{D}_1$ προκύπτει από κανόνες της μορφής $\mathcal{K}, \mathcal{G} \vdash^{\mathcal{I}_{\mathrm{HS}}} \mathcal{D}_1$, όπου οι $\mathcal{K}$ και $\mathcal{G}$ έχουν κάποια από τις παρακάτω μορφές:

   (α′) Η $\mathcal{K}$ έχει τύπο 4.1 (ή 4.2), της μορφής $C(x) \leftarrow R(x, y)$, και η $\mathcal{G}$ έχει τύπο 2.1 (2.2), και είναι της μορφής $R(x, f(x)) \leftarrow D(x)$. Εφόσον η πρόταση $\mathcal{K}$ ανήκει στο $\mathcal{T}$, εαν επιπρόσθετα ανήκει και η $\mathcal{G}$ στο $\mathcal{T}$, τότε μπορούμε να δείξουμε ότι η $\mathcal{D}_1$ μπορεί να παραχθεί από το $\mathcal{I}_{\mathsf{ESLD}}$. Αν η $\mathcal{G}$ παράγεται με υπέρθεση, ή από κανόνα που εφαρμόζεται σε τύπους 3.3, 2.1, όμοια με την προηγούμενη περίπτωση 2, μπορούμε να δείξουμε ότι η $\mathcal{G}$ παράγεται με τον $\mathcal{I}_{\mathsf{ESLD}}$. Συνεπώς, η $\mathcal{D}_1$ είναι δυνατό να παραχθεί από το $\mathcal{I}_{\mathsf{ESLD}}$.

   (β′) Η $\mathcal{K}$ έχει τύπο 3.3 και είναι της μορφής $C(x) \leftarrow D(x) \wedge B(f(x))$, και η $\mathcal{G}$ είναι της μορφής $B(f(x)) \leftarrow K(x)$. Η $\mathcal{K}$ είτε παράγεται από υπέρθεση που εφαρμόζεται σε προτάσεις 2.3, 6.2, και άρα, από την ΕΥ2 η $\mathcal{K}$ παράγεται με τον $\mathcal{I}_{\mathsf{ESLD}}$, είτε παράγεται από κανόνα επίλυσης σε μία RA-clause, με τύπο 4.1 (4.2) και μία πρόταση με τύπο 2.1 (2.2), και έτσι, σύμφωνα με την ΕΥ1, παράγεται από το $\mathcal{I}_{\mathsf{ESLD}}$. Επιπλέον, η $\mathcal{G}$ ανήκει στο $\mathcal{T}$, ή παράγεται από υπέρθεση σε προτάσεις με τύπο 2.3 και 6.1, και επομένως, με βάση την ΕΥ1, παράγεται με τον $\mathcal{I}_{\mathsf{ESLD}}$.

Απόδειξη του Ισχυρισμού 2: Αποδεικνύουμε τον Ισχυρισμό 2 επαγωγικά. Δείχνουμε μόνο την περίπτωση που η $\mathcal{L}$ είναι της μορφής $y \approx z \leftarrow \mathbf{A}(x) \leftarrow R(x, y) \wedge R(x, z) \wedge B(z)$. Η περίπτωση που η $\mathcal{L}$ διαθέτει ανάστροφο ρόλο είναι παρόμοια.

Βασική περίπτωση: Έστω $\Upsilon, \mathcal{D} \vdash \mathcal{F}_1$. Εφόσον οι προτάσεις με τύπο 5 δεν παράγονται ποτέ από το $\mathcal{I}_{\mathrm{HS}}$, η $\mathcal{L}$ είναι στο $\mathcal{T}$. Σύμφωνα με τον Πίνακα 5.2 η $\mathcal{D}$ έχει τύπο 2.1, ή 2.2. Εξετάζουμε διαφορετικές περιπτώσεις για την παραγωγή της $\mathcal{D}$:

1. Η $\mathcal{D}$ παράγεται από κανόνα υπέρθεσης που εφαρμόζει μία πρόταση με τύπο 6.1 Από την ΕΥ1, παράγεται από το $\mathcal{I}_{\mathsf{ESLD}}$.

2. Η $\mathcal{D}$ παράγεται από κανόνα της μορφής $\mathcal{D}_1, \mathcal{J}_1 \vdash \mathcal{D}$, όπου η $\mathcal{D}_1$ έχει τύπο 2.1(2.2), $R(x, f(x)) \leftarrow A(x) \wedge B(f(x))$ και η $\mathcal{J}_1$ έχει τύπο 2.3, $B(f(x)) \leftarrow C(x)$.

Η πρόταση $\mathcal{D}_1$ μπορεί να παραχθεί μόνο από κανόνα υπέρθεση που εφαρμόζει μία πρόταση με τύπο 6.2 και έτσι, από την ΕΥ1, ισχύει ότι η $\mathcal{D}$ παράγεται με τον $\mathcal{I}_{\mathsf{ESLD}}$.

3. Η $\mathcal{D}$ παράγεται από κανόνα της μορφής $\mathcal{E}_1, \mathcal{D}_1 \vdash \mathcal{D}$, όπου η $\mathcal{E}_1$ έχει τύπο 3.1(3.2) και η $\mathcal{D}_1$ έχει τύπο 2.1 (2.2). Συνεπώς, ο κανόνας $\Upsilon, \mathcal{D} \vdash^{\mathcal{I}_{\mathrm{HS}}} \mathcal{F}_1$ μπορεί να εκτυλιχθεί στους κανόνες $\Upsilon, \mathcal{E}_1 \vdash \Upsilon_2$ και $\Upsilon_2, \mathcal{D}_1 \vdash \mathcal{F}_1$. Οι προτάσεις με τύπο 3.1 δεν παράγονται ποτέ από το $\mathcal{I}_{\mathrm{HS}}$, και άρα ισχύει ότι $\mathcal{E}_1 \in \mathcal{T}$, ενώ η $\mathcal{D}_2$ (με τύπο 2.1) παράγεται σε μικρότερο βάθος. Από έναν άμεσο ισχυρισμό δείχνουμε ότι μπορούμε να εκτυλίξουμε εξαντλητικά αυτούς τους κανόνες, μέχρις ότου να προκύψει μία παραγωγή της μορφής $\Upsilon_1, \ldots, \Upsilon_n, \mathcal{F}_1$, όπου $\Upsilon_i, \mathcal{E}_i \vdash \Upsilon_{i+1}$, $\Upsilon_1 = \Upsilon$ και οι Υ_i έχουν τύπο 5, οι $\mathcal{E}_i$ έχουν τύπο 3.1 (ή 3.2) και ανήκουν στο $\mathcal{T}$ και η $\mathcal{F}_1$ παράγεται με τον κανόνα $\Upsilon_n, \mathcal{D}_n \vdash \mathcal{F}_1$, όπου η $\mathcal{D}_n$ έχει τύπο 2.1 (ή 2.2). Η πρόταση $\mathcal{D}_n$ είτε ανήκει στο $\mathcal{T}$, είτε προκύπτει από κανόνα υπέρθεσης και άρα από την ΕΥ1, μπορεί να κατασκευαστεί από το $\mathcal{I}_{\mathsf{ESLD}}$.

Συνεπώς, η πρόταση $\mathcal{F}_1$ παράγεται από την $\mathcal{L}$ με τον $\mathcal{I}_{\mathsf{ESLD}}$.

Επαγωγικό βήμα: Υποθέτουμε ότι για $1 \leq i \leq n$ όλες οι προτάσεις $\mathcal{F}_i$ με τύπο 6.3 που παράγονται από το $\mathcal{I}_{\mathrm{HS}}$ ξεκινώντας από την $\mathcal{L}$ είναι δυνατό να κατασκευαστούν και με τον $\mathcal{I}_{\mathsf{ESLD}}$ (ΕΥ3). Υποθέτουμε ότι μία νέα πρόταση $\mathcal{F}$ με τύπο 6.3 παράγεται με τον $\mathcal{I}_{\mathrm{HS}}$ από έναν κανόνα $\mathcal{F}_n, \mathcal{N} \vdash \mathcal{F}$. Σύμφωνα με τον Πίνακα 5.2 η πρόταση $\mathcal{F}_n$ έχει τύπο 6.3 και από την ΕΥ2 έχουμε ότι παράγεται με τον $\mathcal{I}_{\mathsf{ESLD}}$. Επιπλέον, η πρόταση $\mathcal{N}$ έχει τύπο 2.3 και είναι της μορφής $A(f(x)) \leftarrow B(x)$, χωρίς συναρτήσεις στο σώμα. Μελετάμε τις διαφορετικές περιπτώσεις για την $\mathcal{N}$:

1. Η $\mathcal{N}$ παράγεται από κανόνα υπέρθεσης. Με βάση την ΕΥ1, παράγεται από το $\mathcal{I}_{\mathsf{ESLD}}$.

2. Η $\mathcal{N}$ παράγεται από κανόνα που εφαρμόζεται στις προτάσεις 4.1 (4.2) και 2.2 (2.1). Δείχνουμε ότι η $\mathcal{N}$ κατασκευάζεται με τον $\mathcal{I}_{\mathsf{ESLD}}$ όπως και στην περίπτωση 2b του Ισχυρισμού 2.

3. Η $\mathcal{N}$ παράγεται από κανόνα της μορφής $\mathcal{K}, \mathcal{G} \vdash \mathcal{N}$, όπου η $\mathcal{K}$ έχει τύπο 3.3, της μορφής $C(x) \leftarrow \mathbf{B}(x)$, και την $\mathcal{G}$ που έχει τύπο 2.3, $B(f(x)) \leftarrow D(x)$. Σε αυτή την περίπτωση ο κανόνας $\mathcal{F}_n, \mathcal{N} \vdash \mathcal{F}$ μπορεί να ξεδιπλωθεί στους $\mathcal{F}_n, \mathcal{K} \vdash \mathcal{F}'$ και $\mathcal{F}'_n, \mathcal{G} \vdash \mathcal{F}$. Από έναν άμεσο επαγωγικό ισχυρισμό μπορούμε να εκτυλίξουμε εξαντλητικά αυτούς τους κανόνες μέχρις ότου να προκύψει μία παραγωγή της μορφής $\mathcal{F}_i, \ldots, \mathcal{F}_{i+k}, \mathcal{F}$, όπου οι $\mathcal{F}_j, \mathcal{E}_j \vdash \mathcal{F}_{j+1}$ έχουν τύπο 6.3, οι $\mathcal{E}_j$ έχουν τύπο 3.3 και ανήκουν στο $\mathcal{T}$, ή με βάση τον Ισχυρισμό 1 παράγονται από το $\mathcal{I}_{\mathsf{ESLD}}$, και η $\mathcal{F}$ παράγεται από τον κανόνα $\mathcal{F}_{i+k}, \mathcal{N}_{i+k} \vdash \mathcal{F}$, όπου η $\mathcal{N}_{i+k}$ έχει τύπο 2.3 και παράγεται σε μικρότερο βάθος.

4. Η $\mathcal{N}$ παράγεται από κανόνα της μορφής $\mathcal{K}, \mathcal{G} \vdash \mathcal{N}$, όπου οι $\mathcal{K}$ και $\mathcal{G}$ έχουν τύπο 2.3. Τότε, ο κανόνας $\mathcal{F}_n, \mathcal{N} \vdash^{\mathcal{I}_{\mathrm{HS}}} \mathcal{F}$ μπορεί να εκτυλιχθεί σε κανόνες της μορφής $\mathcal{F}_n, \mathcal{K} \vdash \mathcal{F}'_n$, $\mathcal{F}'_n, \mathcal{G} \vdash \mathcal{F}'$ όπου οι $\mathcal{K}$ και $\mathcal{G}$ παράγονται σε μικρότερα βάθη.

Συνεπώς, η $\mathcal{F}$ έχει τύπο 6.3 και παράγεται από την $\mathcal{L}$ με τον $\mathcal{I}_{\mathsf{ESLD}}$.

Απόδειξη του Ισχυρισμού 3: Χρησιμοποιούμε ξανά επαγωγή.

Βασική περίπτωση: Έστω ότι $\mathcal{F}, \mathcal{D} \vdash^{\mathcal{I}_{\mathrm{HS}}} \Upsilon_1$. Διακρίνουμε διαφορετικές περιπτώσεις:

1. Η πρόταση $\mathcal{F}$ είναι της μορφής $f(z) \approx z \leftarrow \mathbf{A}(x) \wedge R(x,z) \wedge B(z)$ και η $\mathcal{D}$ είναι της μορφής $R(x, g(x)) \leftarrow \mathbf{C}(x)$, με τύπο 2.1. Σε αυτή την περίπτωση η Υ_1 έχει τύπο 6.1. Η πρόταση $\mathcal{D}$ είτε παράγεται από κανόνα υπέρθεσης, οπότε με βάση την ΕΥ1 παράγεται από το $\mathcal{I}_{\mathsf{ESLD}}$, είτε παράγεται από ένα κανόνα της μορφής $\mathcal{E}_1, \mathcal{D}_1 \vdash \mathcal{D}$, όπου η $\mathcal{E}_1$ έχει τύπο 3.1 (3.2), έστω $R(x,y) \leftarrow S(x,y)$, και η $\mathcal{D}_1$ έχει και πάλι τύπο 2.1 (2.2), έστω $S(x, g(x)) \leftarrow C(x)$. Ο κανόνας, $\mathcal{F}, \mathcal{D}_1 \vdash^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{D}$ μπορεί να εκτυλιχθεί στους $\mathcal{F}, \mathcal{E}_1 \vdash \mathcal{F}_1$ και $\mathcal{F}_1, \mathcal{D}_1 \vdash \mathcal{D}$. Οι προτάσεις με τύπο 3.1 δεν παράγονται από το $\mathcal{I}_{\mathrm{HS}}$, και έτσι, ισχύει ότι $\mathcal{E}_1 \in \mathcal{T}$, ενώ η $\mathcal{D}_1$ (με τύπο 2.1) έχει μικρότερο βάθος παραγωγής. Από έναν επαγωγικό ισχυρισμό δείχνουμε ότι εκτυλίσσουμε εξαντλητικά αυτούς τους κανόνες μέχρι να προκύψει μία παραγωγή της μορφής $\mathcal{F}_1, \ldots, \mathcal{F}_n, \mathcal{C}$, όπου $\mathcal{F}_i, \mathcal{E}_i \vdash \mathcal{F}_{i+1}$, οι $\mathcal{F}_i$ έχουν τύπο 6.3, και με βάση τον Ισχυρισμό 2, παράγονται από το $\mathcal{I}_{\mathsf{ESLD}}$, οι $\mathcal{E}_i$ έχουν τύπο 3.1 (ή 3.2) και ανήκουν στο $\mathcal{T}$ και η $\mathcal{C}$ παράγεται από κάποιο κανόνα $\mathcal{F}_n, \mathcal{D}_n \vdash \mathcal{C}$, όπου η $\mathcal{D}_n$ έχει τύπο 2.1 (ή 2.2) και ανήκει στο $\mathcal{T}$, ή έχει προκύψει από υπέρθεση, και έτσι από την ΕΥ1, μπορεί να προκύψει από το $\mathcal{I}_{\mathsf{ESLD}}$. Καταλήγουμε ότι η Υ_1 παράγεται με τον $\mathcal{I}_{\mathsf{ESLD}}$.

2. Η πρόταση $\mathcal{F}$ έχει τη μορφή $f(z) \approx z \leftarrow \mathbf{A}(x) \wedge R(x,z) \wedge B(z)$ και η $\mathcal{D}$ έχει τύπο 2.2, της μορφής $R(g(x), x) \leftarrow \mathbf{C}(x)$. Σε αυτή την περίπτωση η Υ_1 έχει τύπο 6.2. Η πρόταση $\mathcal{D}$ μπορεί είτε να παραχθεί με υπέρθεση, οπότε και με βάση την ΕΥ1, παράγεται από τον $\mathcal{I}_{\mathsf{ESLD}}$, ή η $\mathcal{D}_1$ παράγεται από ένα κανόνα της μορφής $\mathcal{E}_1, \mathcal{D}_1 \vdash \mathcal{D}$, όπου η $\mathcal{E}_1$ έχει τύπο 3.1 (ή 3.2), έστω $R(x,y) \leftarrow S(x,y)$ και η $\mathcal{D}_1$ έχει τύπο 2.2, (2.1), έστω $S(g(x), x) \leftarrow C(x)$. Όπως είναι φανερό, ο $\mathcal{F}, \mathcal{D}_1 \vdash^{\mathcal{I}_{\mathsf{REQ}}} \mathcal{D}$ είναι δυνατό να εκτυλιχθεί στους $\mathcal{F}, \mathcal{E}_1 \vdash \mathcal{F}_1$ και $\mathcal{F}_1, \mathcal{D}_1 \vdash \mathcal{D}$. Οι προτάσεις με τύπο 3.1 δεν παράγονται από το $\mathcal{I}_{\mathrm{HS}}$, και επομένως $\mathcal{E}_1 \in \mathcal{T}$, ενώ η $\mathcal{D}_1$ (και πάλι με τύπο 2.1) έχει μικρότερο βάθος παραγωγής. Από έναν επαγωγικό ισχυρισμό μπορούμε να δείξουμε ότι οι κανόνες εκτυλίσσονται εξαντλητικά, μέχρις ότου προκύψει μία παραγωγή της μορφής $\mathcal{F}_1, \ldots, \mathcal{F}_n, \mathcal{C}$, όπου $\mathcal{F}_i, \mathcal{E}_i \vdash \mathcal{F}_{i+1}$, οι $\mathcal{F}_i$ έχιυν τύπο 6.3 και έτσι παράγονται από το $\mathcal{I}_{\mathsf{ESLD}}$ (Ισχυρισμός 2), οι $\mathcal{E}_i$ έχουν τύπο 3.1 (ή 3.2) και ανήκουν επίστης στο $\mathcal{T}$, και η $\mathcal{C}$ παράγεται από κάποιον κανόνα $\mathcal{F}_n, \mathcal{D}_n \vdash \mathcal{C}$, όπου η $\mathcal{D}_n$ έχει τύπο 2.1 (ή 2.2) και ανήκει στο $\mathcal{T}$, ή παράγεται από το $\mathcal{I}_{\mathsf{ESLD}}$ με υπέρθεση. Συμπεραίνουμε ότι η Υ_1 παράγεται από το $\mathcal{I}_{\mathsf{ESLD}}$.

3. Η πρόταση $\mathcal{F}_1$ έχει τύπο 6.3, της μορφής $x \approx z \leftarrow \mathbf{A}(f(x)) \wedge R(f(x), z) \wedge B(x) \wedge C(z)$ και η $\mathcal{D}_1$ έχει τύπο 2.1, της μορφής $R(x, g(x)) \leftarrow \mathbf{C}(x)$. Η απόδειξη είναι όμοια με την προηγούμενη.

Υποθέτουμε ότι σε κάποιο σημείο οι προτάσεις Υ_i με τύπο 6.1 και 6.2 που παράγονται από το $\mathcal{T}$ με τον $\mathcal{I}_{\mathrm{HS}}$ ξεκινώντας από την πρόταση $\mathcal{F}$ με τύπο 6.3 παράγονται

επίσης από το $\mathcal{T}$ με τον $\mathcal{I}_{\mathsf{ESLD}}$, ξεκινώντας από την πρόταση $\mathcal{F}$ (ΕΥ4). Υποθέτουμε ότι μία νέα πρόταση Υ με τύπο 6.1, ή 6.2, προκύπτει σε κάποια παραγωγή που ξεκινάει με την $\mathcal{F}$. Σύμφωνα με τον Πίνακα 5.2 αυτές οι προτάσεις παράγονται από κανόνες της μορφής $\Upsilon_n, \mathcal{D} \vdash^{\mathcal{I}_{\mathrm{HS}}} \Upsilon$ που έχουν έναν από τους ακόλουθους τύπους:

1. Η πρόταση Υ_n έχει τύπο 6.2 της μορφής $g(f(x)) \approx x \leftarrow A(f(x)) \wedge B(x) \wedge C(g(f(x)))$, ή $g(f(x)) \approx x \leftarrow C(f(x)) \wedge B(x)$, ή τύπο 6.1 της μορφής $g(x) \approx f(x) \leftarrow C(f(x)) \wedge B(x)$. Η $\mathcal{D}$ έχει τύπο 2.3 της μορφής $C(g(x)) \leftarrow K(x)$. Η απόδειξη είναι ίδια για όλες τις περιπτώσεις. Η παραγωγή της πρότασης $\mathcal{D}$ αντιστοιχίζεται σε μία από τις παρακάτω περιπτώσεις:

   (α′) Η $\mathcal{D}$ παράγεται με υπέρθεση σε μία πρόταση με τύπο 2.3 που ανήκει στο $\mathcal{T}$, μίας άλλης με τύπο 6.1. Από την ΕΥ1, η $\mathcal{D}$ παράγεται από το $\mathcal{I}_{\mathsf{ESLD}}$,

   (β′) Η $\mathcal{D}$ παράγεται από ένα κανόνα $\mathcal{J}_1, \mathcal{D}_1 \vdash \mathcal{D}$, όπου η $\mathcal{J}_1$ έχει τύπο 4.1 (4.2) και η $\mathcal{D}_1$ έχει τύπο 2.2 (2.1). Η $\mathcal{J}_1$ ανήκει στο $\mathcal{T}$ αφού ο $\mathcal{I}_{\mathrm{HS}}$ δεν παράγει ποτέ προτάσεις με τύπο 4.1. Η $\mathcal{D}_1$ μπορεί να προκύψει από ένα κανόνα της μορφής $\mathcal{E}_2, \mathcal{D}_2 \vdash \mathcal{D}_1$, όπου η $\mathcal{E}_2$ έχει τύπο 3.1, ή 3.2 και η $\mathcal{D}_2$ έχει τύπο 2.2, 2.1 αντίστοιχα. Προφανώς, ο $\mathcal{J}_1, \mathcal{D}_1 \vdash \mathcal{D}$ μπορεί να εκτυλιχθεί στους κανόνες $\mathcal{J}_1, \mathcal{E}_2 \vdash \mathcal{J}_2$ και $\mathcal{J}_2, \mathcal{D}_2 \vdash \mathcal{D}$. Οι προτάσεις με τύπο 3.1 δεν παράγονται ποτέ από το $\mathcal{I}_{\mathrm{HS}}$, και άρα ισχύει $\mathcal{J}_1 \in \mathcal{T}$, ενώ η $\mathcal{D}_1$ (με τύπο 2.1) έχει μικρότερο βάθος παραγωγής. Και πάλι, μπορούμε να δείξουμε ότι οι κανόνες εκτυλίσσονται εξαντλητικά, μέχρις ότου να προκύψει μία παραγωγή της μορφής $\mathcal{J}_1, \ldots, \mathcal{J}_n, \mathcal{D}$ όπου $\mathcal{J}_i, \mathcal{E}_i \vdash \mathcal{J}_{i+1}$, οι $\mathcal{J}_i$ έχουν τύπο 4.1 (και άρα παράγονται με τον $\mathcal{I}_{\mathsf{ESLD}}$), οι $\mathcal{E}_i$ έχουν τύπο 3.1 (ή 3.2) και ανήκουν στο $\mathcal{T}$ και η $\mathcal{D}$ παράγεται από ένα κανόνα $\mathcal{F}_n, \mathcal{D}_n \vdash \mathcal{C}$, όπου η $\mathcal{D}_n$ έχει τύπο 2.1, ανήκει στο $\mathcal{T}$, ή παράγεται από κανόνα υπέρθεσης και άρα προκύπτει από το $\mathcal{I}_{\mathsf{ESLD}}$ με βάση την ΕΥ1.

   (γ′) Η $\mathcal{D}$ παράγεται από κανόνα της μορφής $\mathcal{E}_1, \mathcal{D}_1 \vdash \mathcal{D}$, όπου η $\mathcal{E}_1$ έχει τύπο 3.3, έστω $C(x) \leftarrow \mathbf{B}(x)$, και η $\mathcal{D}_1$ έχει τύπο 2.3, έστω $B(g(x)) \leftarrow K(x)$. Προφανώς, ο $\mathcal{K}_n, \mathcal{D} \vdash^{\mathcal{I}_{\mathrm{HS}}} \mathcal{K}$ εκτυλίσσεται στους $\mathcal{K}_n, \mathcal{E}_1 \vdash \mathcal{K}'$ και $\mathcal{K}', \mathcal{D}_1 \vdash \mathcal{K}$. Από έναν επαγωγικό ισχυρισμό δείχνουμε ότι οι κανόνες εκτυλίσσονται μέχρι να προκύψει μία παραγωγή $\mathcal{K}_n, \ldots, \mathcal{K}_{n+k}, \mathcal{D}$ όπου $\mathcal{K}_i, \mathcal{E}_i \vdash \mathcal{K}_{i+1}$, οι $\mathcal{K}_i$ έχουν τύπο 6.3 και παράγονται από το $\mathcal{I}_{\mathsf{ESLD}}$, οι $\mathcal{E}_i$ έχουν τύπο 3.3 και είτε ανήκουν στο $\mathcal{T}$, είτε παράγονται από το $\mathcal{I}_{\mathsf{ESLD}}$ σύμφωνα με τον Ισχυρισμό 1, και η $\mathcal{K}$ παράγεται από ένα κανόνα $\mathcal{K}_{n+k}, \mathcal{D}_n \vdash \mathcal{K}$, όπου η $\mathcal{D}_n$ με τύπο 2.3 ανήκει στο $\mathcal{T}$, ή παράγεται με υπέρθεση, και άρα μπορεί να προκύψει από το $\mathcal{I}_{\mathsf{ESLD}}$.

   (δ′) Η $\mathcal{D}$ παράγεται από ένα κανόνα της μορφής $\mathcal{E}, \mathcal{G} \vdash \mathcal{D}$, όπου οι $\mathcal{E}$ και $\mathcal{G}$ έχουν τύπο 2.3. Ο κανόνας $\mathcal{K}_n, \mathcal{D} \vdash^{\mathcal{I}_{\mathrm{HS}}} \mathcal{K}$ μπορεί να εκτυλιχθεί σε δύο κανόνες της μορφής $\mathcal{K}_n, \mathcal{E} \vdash \mathcal{K}'_n$, $\mathcal{K}'_n, \mathcal{G} \vdash \mathcal{K}$ όπου οι $\mathcal{E}$ και $\mathcal{G}$ παράγονται σε μικρότερα βάθη.

Συμπεραίνουμε ότι κάθε πρόταση Υ με τύπο 6.1, 6.2 που παράγεται με τον $\mathcal{I}_{\mathrm{HS}}$ παράγεται επίσης, με τον $\mathcal{I}_{\mathsf{ESLD}}$ από το $\mathcal{T}$.

Θεωρούμε τον κανόνα υπέρθεσης $\mathcal{C}_1, \Upsilon \vdash \mathcal{C}$. Σύμφωνα με τον Ισχυρισμό 2, η δευτερεύουσα προκείμενη Υ παράγεται με τον $\mathcal{I}_{\mathsf{ESLD}}$. Επιπλέον, από τον ορισμό της βασικής υπέρθεσης η πρόταση $\mathcal{C}_1$ ανήκει στο $\mathcal{T}$. Καταλήγουμε ότι η $\mathcal{C}$ παράγεται με το $\mathcal{I}_{\mathsf{ESLD}}$. Αν η Υ έχει τύπο 6.1, τότε η πρόταση $\mathcal{C}$ δε διαθέτει συναρτήσεις στο σώμα και άρα, η $\mathcal{C}'$ συμπίπτει με την $\mathcal{C}$. Απομένει να εξετάσουμε την περίπτωση όπου η πρόταση Υ έχει τύπο 6.2 και η $\mathcal{C}$ περιλαμβάνει συναρτήσεις στο σώμα.

Σύμφωνα με τον $\mathcal{I}_{\mathrm{HS}}$ η $\mathcal{C}'$ παράγεται από ένα κανόνα $\mathcal{C}, \mathcal{D} \vdash^{\mathcal{I}_{\mathrm{HS}}} \mathcal{C}'$ όπου η $\mathcal{D}$ έχει τύπο 2.3 και είναι της μορφής $B(f(x)) \leftarrow D(x)$. Αν η $\mathcal{D}$ ανήκει στο $\mathcal{T}$, τότε η $\mathcal{C}'$ παράγεται επίσης από το $\mathcal{I}_{\mathsf{ESLD}}$. Αν η $\mathcal{D}$ παράγεται από κανόνα υπέρθεσης που εφαρμόζει κάποια πρόταση με τύπο 6.1, τότε από το Ισχυρισμό 2, προκύπτει ότι η $\mathcal{C}'$ μπορεί να παραχθεί από τον $\mathcal{I}_{\mathsf{ESLD}}$. Αν η $\mathcal{D}$ παράγεται απο ένα κανόνα $\mathcal{F}, \mathcal{G} \vdash \mathcal{D}$, τότε δείχνουμε ότι είτε ο κανόνας $\mathcal{C}, \mathcal{D} \vdash \mathcal{C}'$ μπορεί να εκτυλιχθεί στους $\mathcal{C}, \mathcal{F} \vdash \mathcal{C}_1$, $\mathcal{C}_1, \mathcal{G} \vdash \mathcal{C}'$, είτε η $\mathcal{C}'$ παράγεται από το $\mathcal{I}_{\mathsf{ESLD}}$. Στη συνέχεια εξετάζουμε διαφορετικές περιπτώσεις για τις $\mathcal{F}, \mathcal{G}$:

1. Η $\mathcal{F}$ έχει τύπο 4.1, ή 4.2 και η $\mathcal{G}$ έχει τύπο 2.2, ή 2.1. Όπως έχουμε ήδη αποδείξει (Ισχυρισμός 1, περίπτωση 2β) η $\mathcal{C}'$ παράγεται από το $\mathcal{I}_{\mathsf{ESLD}}$.

2. Η $\mathcal{F}$ έχει τύπο 3.3, και είναι της μορφής $B(x) \leftarrow \mathbf{C}(x)$, και η $\mathcal{G}$ με τύπο 2.3 είναι της μορφής $C(f(x)) \leftarrow D(x)$. Προφανώς, ο κανόνας $\mathcal{C}, \mathcal{D} \vdash^{\mathcal{I}_{\mathrm{HS}}} \mathcal{C}'$ εκτυλίσσεται σε μία ακολουθία $\mathcal{C}, \mathcal{F} \vdash \mathcal{C}_1, \mathcal{C}_1, \mathcal{G} \vdash \mathcal{C}'$. Εκτυλίσσουμε εξαντλητικά τους κανόνες μέχρι να κατασκευάσουμε μία παραγωγή της μορφής $\mathcal{C}_1, \ldots, \mathcal{C}_n, \mathcal{C}'$ όπου $\mathcal{C}_i, \mathcal{F}_i \vdash \mathcal{C}_{i+1}$, οι προτάσεις $\mathcal{C}_i$ έχουν τύπο 2.1, ή 2.2, οι $\mathcal{F}_i$ έχουν τύπο 3.3 στο $\mathcal{T}$, ή παράγονται από το $\mathcal{I}_{\mathsf{ESLD}}$ σύμφωνα με τον Ισχυρισμό 1. Η $\mathcal{C}'$ παράγεται από κάποιο κανόνα $\mathcal{C}_n, \mathcal{D}_n \vdash \mathcal{C}$, όπου η $\mathcal{D}_n$ έχει τύπο 2.1 και ανήκει στο $\mathcal{T}$, ή προκύπτει από υπέρθεση και άρα, σύμφωνα με την ΕΥ1, παράγεται από το $\mathcal{I}_{\mathsf{ESLD}}$.

3. Η $\mathcal{F}$ έχει τύπο 2.3 της μορφής $B(f(x)) \leftarrow K(x) \wedge C(f(x))$ και η $\mathcal{G}$ έχει τύπο 2.3, της μορφής $C(f(x)) \leftarrow D(x)$. Ο κανόνας $\mathcal{C}, \mathcal{D} \vdash^{\mathcal{I}_{\mathrm{HS}}} \mathcal{C}'$ εκτυλίσσεται σε μία ακολουθία $\mathcal{C}, \mathcal{F} \vdash \mathcal{C}_1, \mathcal{C}', \mathcal{G} \vdash \mathcal{C}'$.

Καταλήγουμε ότι η $\mathcal{C}'$ παράγεται από το $\mathcal{I}_{\mathsf{ESLD}}$.

□

Λήμμα 5.3.3. *Έστω $\mathcal{T}$ ένα Horn-$\mathcal{SHIQ}$ TBox και $\mathcal{Q}$ ένα ΣΕ. Κάθε πρόταση με τύπο 1 που παράγεται από το $\mathcal{T}$ με τον $\mathcal{I}_{\mathrm{HS}}$ ξεκινώντας από το $\mathcal{Q}$, παράγεται επίσης από το $\mathcal{T}$, με τον $\mathcal{I}_{\mathsf{ESLD}}$ ξεκινώντας από το $\mathcal{Q}$.*

Απόδειξη. Σϋμφωνα με τον ορισμό του ΣΕ μία πρόταση $\mathcal{Q}$ με τύπο 1 που ανήκει στο $\mathcal{T}$, δεν περιλαμβάνει συναρτήσεις. Επομένως, ένας βασικός κανόνας υπέρθεσης δεν εφαρμόζεται σε μία πρόταση με τύπο 1.

Υποθέτουμε ότι κάθε πρόταση $\mathcal{Q}_i$ με τύπο 1 που παράγεται από το $\mathcal{I}_{\mathrm{HS}}$ ξεκινώντας με το $\mathcal{Q}$ παράγεται επίσης με τον $\mathcal{I}_{\mathsf{ESLD}}$. Έστω $\mathcal{Q}_i, \mathcal{C} \vdash^{\mathcal{I}_{\mathrm{HS}}} \mathcal{Q}_{i+1}$, σύμφωνα με τον Πίνακα 5.2 το συμπέρασμα $\mathcal{Q}_{i+1}$ έχει επίσης τύπο 1, και η πρόταση $\mathcal{C}$ έχει τύπο 2.1, 2.2, ή 2.3, ενώ το σώμα της δε διαθέτει συναρτήσεις. Ενδιαφερόμαστε για την περίπτωση που κατά την παραγωγή της $\mathcal{C}$ εμπλέκεται κάποιος κανόνας υπέρθεσης. Σε κάθε άλλη

περίπτωση, η απόδειξη συμπίπτει με την απόδειξη του λήμματος 4.2.2 για την $\mathcal{ELHI}$. (Σημειώνουμε ότι ο λογισμός $\mathcal{I}_{\mathrm{HS}}$ συμπίπτει με τον $\mathcal{I}_{\mathsf{REQ}}$ για το $\mathcal{ELHI}$ υποσύνολο της Horn-$\mathcal{SHIQ}$). Σύμφωνα με το Λήμμα 5.3.2, αν η παραγωγή της $\mathcal{C}$ εμπλέκει κανόνα υπέρθεσης, τότε η $\mathcal{C}$ παράγεται επίσης από το $\mathcal{I}_{\mathsf{ESLD}}$. □

Θεώρημα 5.3.4. *Έστω ένα Horn-$\mathcal{SHIQ}$, $\mathcal{T}$ και ένα ΣΕ $\mathcal{Q}$. Κάθε παραγωγή από το $\mathcal{T} \cup \{\mathcal{Q}\}$ με τον $\mathcal{I}_{\mathrm{RHS}}$ τερματίζει. Επίσης, το* DD(Rapid-Hshiq($\mathcal{Q}, \mathcal{T}$)) *είναι μία datalog επαναγραφή του $\mathcal{Q}$ με βάση το $\mathcal{T}$.*

Απόδειξη. Έστω $\Re =$ DD(Rapid-Hshiq($\mathcal{Q}, \mathcal{T}$)). Αποδεικνύουμε την ορθότητα. Πρέπει να δείξουμε ότι για κάθε Horn-$\mathcal{SHIQ}$ ABox $\mathcal{A}$ ισχύει ότι cert($\mathcal{Q}, \mathcal{T}, \mathcal{A}$) = cert($\mathcal{Q}, \Re, \mathcal{A}$).

Εφόσον ο λογισμός $\mathcal{I}_{\mathrm{RHS}}$ βασίζεται στην επίλυση, ισχύει ότι cert($\mathcal{Q}, \mathcal{T}, \mathcal{A}$) $\supseteq$ cert($\mathcal{Q}, \Re, \mathcal{A}$). Προκειμένου να δείξουμε ότι cert($\mathcal{Q}, \mathcal{T}, \mathcal{A}$) $\subseteq$ cert($\mathcal{Q}, \Re, \mathcal{A}$) θα αποδείξουμε ότι κάθε $\mathcal{Q}'$ που παράγεται από το λογισμό $\mathcal{I}_{\mathrm{HS}}$, παράγεται επίσης από το $\mathcal{I}_{\mathrm{RHS}}$. Σύμφωνα με το Λήμμα 5.3.3 κάθε $\mathcal{Q}'$ που παράγεται από το $\mathcal{T} \cup \{\mathcal{Q}\}$ με το λογισμό $\mathcal{I}_{\mathrm{HS}}$, μπορεί επίσης να προκύψει από το $\mathcal{T} \cup \{\mathcal{Q}\}$ με το $\mathcal{I}_{\mathsf{ESLD}}$. Επιπλέον, η παραγωγή της $\mathcal{Q}'$ μπορεί να μετατραπεί σε παραγωγή συμπαγή ως προς τη συνάρτηση σύμφωνα με την Πρόταση 3.2.3.

Υποθέτουμε ότι όλες οι προτάσεις $\mathcal{Q}_i$ χωρίς συνάρτηση με τύπο 1, που παράγονται από το $\mathcal{I}_{\mathsf{ESLD}}$, παράγονται επίσης από το $\mathcal{I}_{\mathrm{RHS}}$. Υποθέτουμε ότι παράγεται μία νέα πρόταση $\mathcal{Q}_{i+1}$ χωρίς συναρτήσεις με τύπο 1. Διακρίνουμε τις παρακάτω 2 περιπτώσεις:

1. Η $\mathcal{Q}_{i+1}$ παράγεται με τον $\mathcal{I}_{\mathsf{ESLD}}$ από κάποια πρόταση $\mathcal{Q}_j$, χωρίς συναρτήσεις με $j \leq i$, και δευτερεύουσα προκείμενη $\mathcal{C}_j$, με τύπο 3.1-3.3. Εφόσον οι $\mathcal{Q}_j$, $\mathcal{Q}_{i+1}$ δε διαθέτουν συναρτήσεις, ο κανόνας που παράγει την $\mathcal{Q}_{i+1}$ αντιστοιχίζεται στον κανόνα εκτύλιξης. Επομένως, $\mathcal{Q}_j, \mathcal{C}_j \vdash^{\mathcal{I}_{\mathrm{RHS}}} \mathcal{Q}_{i+1}$ και άρα $\mathcal{Q}_{i+1} \in \Re$.

2. Η $\mathcal{Q}_{i+1}$ παράγεται από κάποια πρόταση $\mathcal{Q}_j$, με το $\mathcal{I}_{\mathsf{ESLD}}$, μέσω μίας επεκταμένης-SLD παραγωγής της μορφής $\mathcal{Q}_j, \mathcal{Q}_{j+1}, \ldots, \mathcal{Q}_{j+k}$, όπου $\mathcal{Q}_{j+k} = \mathcal{Q}_{i+1}$, όλες οι ενδιάμεσες προτάσεις περιλαμβάνουν κάποιο όρο $f(s_j)$. Εφόσον η παραγωγή είναι συμπαγής ως προς τη συνάρτηση, όλες οι δευτερεύουσες προκείμενες περιλαμβάνουν έναν όρο $f(t_j)$, και είτε είναι της μορφής $R_1(x, f(x)) \leftarrow \mathbf{D}_1, \ldots R_k(x, f(x)) \leftarrow \mathbf{D}_k(x)$ ή $A_1(f(x)) \leftarrow \mathbf{C}_1, \ldots A_k(f(x)) \leftarrow \mathbf{C}_k$ (για τις οποίες έχουμε δείξει ότι παράγονται με το $\mathcal{I}_{\mathsf{ESLD}}$), ενώ όλες οι ενδιάμεσες προτάσεις με τύπο 1 περιλαμβάνουν άτομα της μορφής $R_j(u_j, f(s_j))$, $R_j(f(s_j), u_j)$, ή $A_j(f(s_j))$, όπου τα s_j, u_j είναι όροι με $1 \leq j \leq k$. Επιπλέον, εφόσον η $\mathcal{Q}_j$ δε διαθέτει συναρτήσεις και η $\mathcal{Q}_{j+1}$ αναφέρει το σύμβολο f, το άτομο της $\mathcal{Q}_j$ που αναλύεται με την $\mathcal{C}_j$ είναι της μορφής $R_j(s, t)$, ή $A_j(t)$, και ο μκε περιλαμβάνει μία αντιστοίχιση της μορφής $t \mapsto f(x)$ όπου η t είναι δεσμευμένη μεταβλητή του $\mathcal{Q}_j$.

   Θεωρούμε τα σύνολα $S_1^1, \ldots, S_k^1$ που περιλαμβάνουν όλα τα άτομα ρόλου του $\mathcal{Q}_j$ που επιλέγονται κατά την εφαρμογή των κανόνων επίλυσης και $S_1^2, \ldots, S_k^2$ τα αντίστοιχα σύνολα που περιλαμβάνουν άτομα έννοιας (σημειώνουμε ότι τα σύνολα μπορεί να είναι κενά). Εφόσον υπάρχει ένας μκε για όλες τις ενδιάμεσες εφαρμογές κανόνων κατά την SLD παραγωγή, υπάρχει και ένας άμεσος μκε για

τα $S_j^1 \cup \{R_j(x, f(x))\}$ και $S_j^2 \cup \{A_j(f(x)\}$, $1 \leq j \leq k$ [48]. Επιπλέον, όπως αναλύσαμε παραπάνω, και από την κατασκευή του μκε, ο μκε πρέπει να περιλαμβάνει μία αντιστοίχιση της μορφής $x \mapsto f(y')$ όπου x είναι μία δεσμευμένη μεταβλητή (και όχι μεταβλητή απάντησης). Συνεπώς, είναι δυνατό να εφαρμοστεί ένας κανόνας επίλυσης της μορφής $\mathcal{Q}_j, \mathcal{C}_1, \ldots, \mathcal{C}_n \vdash \mathcal{Q}_{i+1}$ που αντιστοιχίζεται στον κανόνα συρρίκνωσης.

Απομένει να δείξουμε ότι οι δευτερεύουσες προκείμενες που παράγονται από το $\mathcal{I}_{\mathsf{ESLD}}$, είναι επίσης δυνατό να παραχθούν και από το $\mathcal{I}_{\mathrm{RHS}}$. Προφανώς, ένας κανόνας υπέρθεσης $\mathcal{I}, \mathcal{M} \vdash \mathcal{C}$ όπου η $\mathcal{M}$ έχει τύπο 6.1, μπορεί να αντιστοιχιστεί στον sup-1 κανόνα. Στην περίπτωση που η $\mathcal{M}$ έχει τύπο 6.2, όπως έχουμε δείξει στο Λήμμα 5.3.2, είναι δυνατό να παραχθεί μία πρόταση $\mathcal{C}'$ χωρίς συναρτήσεις, με το λογισμό $\mathcal{I}_{\mathsf{ESLD}}$, ξεκινώντας από την πρότση $\mathcal{C}$. Σύμφωνα με το Λήμμα 5.3.2 η πρόταση $\mathcal{C}'$ μπορεί να παραχθεί από την $\mathcal{C}_1$ με τον κανόνα sup-2.

Τέλος, από τον Ισχυρισμό 2 στο Λήμμα 5.3.2 και την Πρόταση 4.2.3, προκύπτει ότι η πρόταση $\mathcal{M}$ με τύπο 6.1, ή 6.2 μπορεί να προκύψει με τους κανόνες εκτύλιξης και συρίκνωσης, ξεκινώντας από την $\mathcal{F}$ με τύπο 6.3. Επιπλέον, από τον Ισχυρισμό 3 στο Λήμμα 5.3.2 και την Πρόταση 4.2.3, έχουμε ότι η $\mathcal{F}$ παράγεται με τους κανόνες εκτύλιξης και συρρίκνωσης ξεκινώντας από την πρόταση $\mathcal{L}$ με τύπο 5.

Συμπεραίνουμε ότι όλες οι προτάσεις με τύπο 1 που δεν περιλαμβάνουν συναρτήσεις είναι δυνατό να παραχθούν από το $\mathcal{I}_{\mathrm{RHS}}$.

□

Μέρος III

Υλοποίηση

Κεφάλαιο 6

Βελτιστοποιήσεις και υλοποίηση

Στην παρούσα ενότητα παρουσιάζουμε τεχνικές βελτιστοποιήσης που αφορούν την υλοποίηση πρακτικού αλγορίθμου που βασίζεται στον $\mathcal{I}_{lite}$.

6.1 Σύνολα εκτύλιξης

Σε πολλές περιπτώσεις, ο κανόνας της εκτύλιξης κατασκευάζει προτάσεις που υπάγονται σε άλλες. Επιπλέον, είναι πιθανό η ίδια πρόταση να παραχθεί περισσότερες από μία φορές.

Παράδειγμα 6.1.1. Έστω $\mathcal{T}$ ένα σώμα ορολογίας που αποτελείται από τις ακόλουθες προτάσεις:

$$R(x, f(x)) \leftarrow A(x) \tag{6.1}$$
$$C(f(x)) \leftarrow A(x) \tag{6.2}$$
$$D(x) \leftarrow C(x) \tag{6.3}$$
$$S(x, y) \leftarrow R(x, y) \tag{6.4}$$

και ένα ερώτημα $\mathcal{Q}_1$, $Q(x) \leftarrow A(x) \wedge S(x, y) \wedge D(y)$. Κατά την εφαρμογή του $\mathcal{I}_{lite}$ στο $\mathcal{T} \cup \{\mathcal{Q}_1\}$ προκύπτουν οι παρακάτω προτάσεις:

$$\mathcal{Q}_1, (6.3) \ \vdash \ Q(x) \leftarrow A(x) \wedge S(x, y) \wedge C(y) \tag{6.5}$$
$$\mathcal{Q}_1, (6.4) \ \vdash \ Q(x) \leftarrow A(x) \wedge R(x, y) \wedge D(y) \tag{6.6}$$
$$(6.5), (6.4) \ \vdash \ Q(x) \leftarrow A(x) \wedge R(x, y) \wedge C(y) \tag{6.7}$$
$$(6.6), (6.3) \ \vdash \ (6.7)$$
$$(6.7), (6.1), (6.2) \ \vdash \ Q(x) \leftarrow A(x) \tag{6.8}$$

Το σύνολο που περιλαβάνει τα $\mathcal{Q}_1$ και (6.5)–(6.8) αποτελεί μία επαναγραφή για το $\mathcal{Q}_1$ με βάση το $\mathcal{T}$. Αρχικά παρατηρούμε ότι το ότι η (6.7) παράγεται δύο φορές. Επιπλέον, επαληθεύεται ότι το ερώτημα (6.8) υπάγει τα $\mathcal{Q}_1$ και (6.5)–(6.7).

Ένα σύστημα που έχει ως στόχο να κατασκευάσει μία επαναγραφή με το μικρότερο δυνατό πλήθος προτάσεων, ιδανικά, θα κατασκεύαζε το ερώτημα (6.8) απευθείας από το $\mathcal{Q}_1$, και έπειτα, εφόσον το (6.8) υπάγει το $\mathcal{Q}_1$, θα έπρεπε το $\mathcal{Q}_1$ να παραλείπεται, ώστε να αποφευχθεί η παραγωγή των ερωτημάτων (6.5)–(6.7). $\diamondsuit$

Όπως μπορεί να γίνει κατανοητό, η επίδοση των συστημάτων επαναγραφής επηρεάζεται αρνητικά, ιδιαίτερα σε περιπτώσεις μεγάλων οντολογιών, όταν η ίδια πρόταση παράγεται περισσότερες φορές, καθώς και όταν παράγονται προτάσεις που υπάγονται σε άλλες. Οι προτάσεις αυτές μπορούν να παραληφθούν από το σύνολο της επαναγραφής, ώστε να έχει μικρότερο μέγεθος και να είναι πιο εύκολη η αποτίμησή της. Εάν μάλιστα, ο αλγόριθμος επαναγραφής δεν κατασκευάσει αυτές τις προτάσεις, η διαδικασία παραγωγής της επαναγραφής διευκολύνεται σημαντικά.

Σαν μία γενική παρατήρηση, μπορούμε να σημειώσουμε ότι τα ερωτήματα που παράγονται από τον κανόνα της συρρίκνωσης είναι πιθανό να υπάγουν ερωτήματα που παράγονται από τον κανόνα της εκτύλιξης. Διαισθητικά, κάτι τέτοιο γίνεται κατανοητό αν λάβουμε υπόψη ότι ο κανόνας της συρρίκνωσης απομακρύνει μια δεσμευμένη μεταβλητή, και επομένως, τα ερωτήματα που παράγει περιλαμβάνουν λιγότερες μεταβλητές από εκείνα που παράγει ο κανόνας της εκτύλιξης. Την ίδια στιγμή όμως, ο κανόνας της συρρίκνωσης δεν μπορεί να εφαρμοστεί, παραμόνο έπειτα από αρκετές εφαρμογές του κανόνα της εκτύλιξης. Πιο συγκεκριμένα, στο προηγούμενο παράδειγμα το ερώτημα (6.8) μπορεί να παραχθεί μόνο μετά την παραγωγή του ερωτήματος (6.7) από τον κανόνα της εκτύλιξης.

Λαμβάνοντας υπόψη τα παραπάνω, η πρακτική υλοποίηση του $\mathcal{I}_{lite}$, συγκεντρώνει την πληροφορία που είναι απαραίτητη για την εφαρμογή του κανόνα της συρρίκνωσης, χωρίς προηγουμένως να εφαρμόσει τον κανόνα της εκτύλιξης, όταν είναι γνωστό εκ των προτέρων ότι η εκτύλιξη θα παράγει προτάσεις που υπάγονται σε άλλες. Επεξηγούμε την προσέγγιση αυτή στον παρακάτω ορισμό.

Ορισμός 6.1.2. Έστω $\mathcal{T}$ ένα σώμα ορολογίας σε DL-Lite, $\mathcal{Q}$ ένα ΣΕ, και έστω A κάποιο άτομο στο σώμα του $\mathcal{Q}$. Έστω επίσης $\mathcal{Q}_A$ ένα ερώτημα τέτοιο ώστε $\mathsf{body}(\mathcal{Q}_A) = \{A\}$ και $\mathsf{avar}(\mathcal{Q}_A) = \mathsf{var}(A) \cap \mathsf{bound}(\mathcal{Q})$. Το σύνολο *εκτύλιξης* του A με βάση τα $\mathcal{Q}, \mathcal{T}$ είναι το σύνολο που περιλαμβάνει όλα τα $\mathsf{body}(\mathcal{Q}')$ τέτοια ώστε το $\mathcal{Q}'$ να παράγεται από το $\mathcal{T} \cup \{\mathcal{Q}_A\}$ χρησιμοποιώντας μόνο τον κανόνα της εκτύλιξης. $\triangle$

Όπως μπορεί να γίνει κατανοητό από την παρακάτω Πρόταση 6.1.3, τα σύνολα εκτύλιξης των ατόμων ενός ερωτήματος $\mathcal{Q}$, χαρακτηρίζουν πλήρως τα ερωτήματα που προκύπτουν από το $\mathcal{Q}$ μέσω του κανόνα της εκτύλιξης.

Πρόταση 6.1.3. *Έστω $\mathcal{T}$ ένα σώμα ορολογίας σε DL-Lite, $\mathcal{Q}$, ένα ΣΕ με σώμα που αποτελείται από τα $A_1, \ldots, A_n$, και έστω $S_1, \ldots, S_n$, τα αντίστοιχα σύνολα εκτύλιξης. Ένα ερώτημα $\mathcal{Q}'$ μπορεί να παραχθεί από το $\mathcal{T} \cup \{\mathcal{Q}\}$ με εφαρμογή του κανόνα εκτύλιξης αν για κάθε S_i υπάρχει $A \in S_i$ τέτοιο ώστε $A \in \mathsf{body}(\mathcal{Q}')$.*

Παράδειγμα 6.1.4. Θεωρούμε το $\mathcal{T}$ και το ερώτημα $\mathcal{Q}_1$ από το Παράδειγμα 6.1.1. Το σύνολο εκτύλιξης S_D του $D(y)$ με βάση το $\mathcal{Q}_1$, περιλαμβάνει όλα τα άτομα

στο σώμα των ερωτημάτων που παράγονται με τον κανόνα εκτύλιξης από το $\mathcal{T} \cup \{Q_D(y) \leftarrow D(y)\}$, δηλαδή τα άτομα στο σώμα των ερωτημάτων $Q_D(y) \leftarrow D(y)$ και $Q_D(y) \leftarrow C(y)$. Παρόμοια, το σύνολο εκτύλιξης S_S του $S(x, y)$ με βάση τα $\mathcal{Q}_1, \mathcal{T}$ περιλαμβάνει τα άτομα στο σώμα των $Q_S(x, y) \leftarrow S(x, y)$ και $Q_S(x, y) \leftarrow R(x, y)$, ενώ το σύνολο εκτύλιξης S_A του $A(x)$ περιλαμβάνει μόνο το $A(x)$.

Είναι φανερό, ότι όλα τα ερωτήματα (6.5)–(6.7) του Παραδείγματος 6.1.1 μπορούν να παραχθούν από τα σύνολα εκτύλιξης. Για παράδειγμα, το ερώτημα (6.7) μπορεί να κατασκευαστεί από το άτομο $A(x)$ από το S_A, το άτομο $R(x, y)$ από το S_S, και το $C(y)$ από το S_D. ♢

Ο αλγόριθμος μας χρησιμοποιεί την πληροφορία που κωδικοποιείται στα σύνολα εκτύλιξης, προκειμένου να κατασκευάσει απευθείας ερωτήματα τα οποία σε διαφορετική περίπτωση, θα προέκυπταν έπειτα από την εφαρμογή περισσότερων κανόνων, αποφεύγοντας έτσι, την εκτέλεση ενδιάμεσων βημάτων.

Παράδειγμα 6.1.5. Θεωρούμε πάλι το Παράδειγμα 6.1.1 και το ερώτημα $\mathcal{Q}_1$. Τα άτομα $S(x, y)$ και $D(y)$ αναφέρουν μεταβλητή y, ενώ το σύνολο εκτύλιξης του $S(x, y)$ περιλαμβάνει το $R(x, y)$ και το σύνολο εκτύλιξης του $D(y)$ περιλαμβάνει το $C(y)$. Επιπλέον, το σώμα ορολογίας περιλαμβάνει τις προτάσεις $R(x, f(x)) \leftarrow A(x)$ και $C(f(x)) \leftarrow A(x)$. Μπορούμε να συμπεράνουμε ότι υπάρχει ερώτημα που κατασκευάζεται έπειτα από περισσότερα βήματα εκτύλιξης (το ερώτημα (6.7) στο Παράδειγμα 6.1.1) επί του οποίου μπορεί να εφαρμοστεί ο κανόνας της συρρίκνωσης με δευτερεύουσες προτάσεις τις (6.1) και (6.2). Έτσι, από το $\mathcal{Q}_1$ μπορούμε άμεσα να κατασκευάσουμε το ερώτημα (6.7), αντικαθιστώντας τα $S(x, y)$ και $D(y)$ με το άτομο $A(x)$. Τέλος, ο αλγόριθμος μπορεί να εντοπίσει ότι το ερώτημα (6.7) υπάγει το $\mathcal{Q}_1$ και έτσι, δεν παράγεται κανένα από τα ερωτήματα (6.5)–(6.7). ♢

Σημειώνουμε ότι ο υπολογισμός των συνόλων εκτύλιξης είναι στις περισσότερες περιπτώσεις, περισσότερο αποδοτικός από ότι ο άμεσος υπολογισμός των ερωτημάτων που προκύπτουν από τον κανόνα εκτύλιξης. Πρώτον, τα ερωτήματα στο σύνολο εκτύλιξης περιέχουν λιγότερα άτομα και έτσι, οι κανόνες εφαρμόζονται πιο αποδοτικά. Δεύτερον, με χρήση κατάλληλης μετονομασίας μεταβλητών τα σύνολα εκτύλιξης μπορούν να επαναχρησιμοποιηθούν. Για παράδειγμα, το σύνολο εκτύλιξης ενός ατόμου $D(y)$ μπορεί να χρησιμοποιηθεί για τον υπολογισμό του συνόλου εκτύλιξης του $D(z)$ μετονομάζοντας τη μεταβλητή y σε z. Τρίτον, αποφεύγοντας την κατασκευή ερωτημάτων καταφέρνουμε να αποφύγουμε περιττά ερωτημάτα, με την έννοια ότι υπάγονται σε άλλα, όπως παρουσιάσαμε στο Παράδειγμα 6.1.1.

Στην περίπτωση που ένα ερώτημα, $\mathcal{Q}$ δεν υπάγεται σε κάποιο που προέκυψε από του κανόνα συρρίκνωσης, τότε ο αλγόριθμος μπορεί να χρησιμοποιήσει τα σύνολα εκτύλιξης προκειμένου να κατασκευάσει όλα τα υπόλοιπα ερωτήματα, όπως φαίνεται στο Παράδειγμα 6.1.4. Παρόλα αυτά, κάτι τέτοιο μπορεί να δημιουργήσει εκθετικό αριθμό ερωτημάτων μιας και πρέπει να θεωρηθούν όλοι οι πιθανοί συνδυασμοί των ατόμων από τα σύνολα εκτύλιξης. Εναλλακτικά, το σύστημα μπορεί να παρέχει μια περισσότερο συμπαγή αναπαράσταση της επαναγραφής κωδικοποιώντας την πληροφορία των συνόλων εκτύλιξης σε κανόνες datalog. Σημειώνουμε ότι παρόμοια

προσέγγιση προτείνεται στο [116]. Στο Παράδειγμα 6.1.4 το σύνολο εκτύλιξης του $D(y)$ περιλαμβάνει το $C(y)$. Έτσι, αντί να χρησιμοποιηθεί το $C(y)$ για την κατασκευή νέων ερωτημάτων από το $\mathcal{Q}_1$, ο αλγόριθμος μπορεί ισοδύναμα, να επιστρέψει τον datalog κανόνα $r_1 = D(x) \leftarrow C(x)$. Παρόμοια, λαμβάνοντας υπόψη το σύνολο εκτυλιξης του ατόμου $S(x, y)$ ο αλγόριθμος μπορεί να επιστρέψει τον datalog κανόνα $r_2 = S(x, y) \leftarrow R(x, y)$. Επιβεβαιώνεται ότι το $\{\mathcal{Q}_1, r_1, r_2\}$ είναι μια datalog επαναγραφή του $\mathcal{Q}_1$ με βάση το $\mathcal{T}$.

Σε ορισμένες περιπτώσεις ενδεχομένως, να χρειάζεται να υπολογιστούν όλα τα ερωτήματα από τα σύνολα εκτύλιξης, όπως για παράδειγμα στην περίπτωση που η επαναγραφή ζητείται στη μορφή ΣΣΕ. Η πληροφορία όμως των συνόλων εκτύλιξης μπορεί να αξιοποιηθεί και για τον εντοπισμό περιττών ερωτημάτων, ώστε αυτά να αγνοηθούν. Εξηγούμε στο παρακάτω παράδειγμα.

Παράδειγμα 6.1.6. Θεωρούμε το ερώτημα $\mathcal{Q}_1 = Q(x) \leftarrow A(x) \wedge D(x)$ και τα σύνολα εκτύλιξης $\{A(x), E(x)\}$ και $\{D(x), E(x)\}$ των ατόμων $A(x)$ και $D(x)$, αντίστοιχα. Με βάση τα σύνολα εκτύλιξης, τα ερωτήματα που προκύπτουν από τον κανόνα της εκτύλιξης είναι μεταξύ άλλων, τα $\mathcal{Q}_2 = Q(x) \leftarrow E(x) \wedge D(x)$, $\mathcal{Q}_3 = Q(x) \leftarrow A(x) \wedge E(x)$, και $\mathcal{Q}_4 = Q(x) \leftarrow E(x)$. Όπως είναι φανερό, το ερώτημα $\mathcal{Q}_4$ υπάγει τα $\mathcal{Q}_2$ και $\mathcal{Q}_3$. Διαισθητικά, αυτό συμβαίνει επειδή και τα δύο σύνολα εκτύλιξης περιλαμβάνουν το άτομο $E(x)$. Έτσι, κάθε ερώτημα που κατασκευάζεται επιλέγοντας από το πρώτο σύνολο εκτύλιξης το άτομο $E(x)$, και διαφορετικό άτομο από το δεύτερο σύνολο εκτύλιξης, θα υπάγεται στο ερώτημα που κατασκευάζεται επιλέγοντας το $E(x)$ και από τα δύο σύνολα εκτύλιξης.

Η ιδέα του προηγούμενου παραδείγματος παρουσιάζεται τυπικά στην επόμενη πρόταση που μπορεί να χρησιμοποιηθεί από τον αλγόριθμο επαναγραφής, ώστε να αποφευχθεί η παραγωγή προτάσεων που υπάργονται σε άλλες.

Πρόταση 6.1.7. *Έστω $\mathcal{T}$ ένα σώμα ορολογίας σε DL-Lite, ένα ΣΕ $\mathcal{Q}$ με σώμα που περιλαμβάνει τα άτομα $A_1, \ldots, A_n$, και έστω, $S_1, \ldots, S_n$ τα αντίστοιχα σύνολα εκτύλιξης. Έστω ένα ερώτημα $\mathcal{Q}'$ που προκύπτει με εκτύλιξη από το $\mathcal{T} \cup \{\mathcal{Q}\}$. Εάν υπάρχουν άτομα $\{B_i, B_j\} \subseteq \mathsf{body}(\mathcal{Q}')$, ένα σύνολο εκτύλιξης S_i τέτοιο ώστε $\{B_i, B_i'\} \subseteq S_i$, και μία αντιστοίχιση $\theta : \mathsf{var}(B_i') \setminus \mathsf{bound}\mathcal{Q} \mapsto \mathsf{var}(\mathcal{Q}')$, τέτοια ώστε $B_i'\theta = B_j$, τότε υπάρχει ένα ερώτημα Q'' που προκύπτει με εκτύλιξη από το $\mathcal{T} \cup \mathcal{Q}$ το οποίο υπάγει το $\mathcal{Q}'$.*

Απόδειξη. Έστω το $\mathcal{Q}'$ που παράγεται με εκτύλιξη από το $\mathcal{T} \cup \{\mathcal{Q}\}$. Υποθέτουμε ότι για κάποια $\{B_k, B_\ell\} \subseteq \mathsf{body}(\mathcal{Q}')$, υπάρχει σύνολο εκτύλιξης S_k με άτομα $\{B_k, B_k'\} \in S_k$, και επίσης υπάρχει μία αντικατάσταση $\theta : \mathsf{var}(B_k') \setminus \mathsf{bound}(\mathcal{Q}) \mapsto \mathsf{var}(\mathcal{Q}')$ τέτοια ώστε $B_k'\theta = B_\ell$.

Θεωρούμε το ερώτημα $\mathcal{Q}''$ που κατασκευάζεται από το $\mathcal{Q}'$ αντικαθιστώντας το άτομο B_k με το B_k'. Από τον ορισμό των συνόλων εκτύλιξης, προκύπτει έυκολα ότι $\mathsf{bound}(\mathcal{Q}'') \subseteq \mathsf{bound}(\mathcal{Q})$. Έτσι, η θ αντιστοιχίζει μόνο μεταβλητές του B_k'. Συνεπώς, για κάθε $i \neq k$ έχουμε ότι $B_i\theta = B_i$, ενώ για το B_k' έχουμε $B_k'\theta = B_\ell$. Έτσι, $\mathcal{Q}''\theta \subseteq \mathcal{Q}'$. Τελικά, από την Πρόταση 6.1.3 προκύπτει ότι το $\mathcal{Q}''$ μπορεί να παραχθεί από το $\mathcal{T} \cup \{\mathcal{Q}\}$ με εκτύλιξη.

□

Η Πρόταση 6.1.7 παρέχει ικανές συνθήκες προκειμένου ο αλγόριθμος επαναγραφής να αποφανθεί εαν ένα ερώτημα που παράγει υπάγεται σε κάποιο άλλο. Το επόμενο παράδειγμα καταδεικνύει πως ένα ερώτημα μπορεί να είναι υπάγεται σε κάποιο άλλο, χωρίς να ικανοποιεί τις συνθήκες της παραπάνω πρότασης.

Παράδειγμα 6.1.8. Έστω ένα ερώτημα $\mathcal{Q} = Q(x) \leftarrow R(x,y) \wedge A(y) \wedge P(x,z) \wedge A(z)$, ένα σώμα ορολογίας $\mathcal{T}$ και τα σύνολα εκτύλιξης $\{R(x,y), P(x,y)\}$ και $\{P(x,z), R(x,z)\}$ για τα άτομα $R(x,y)$ και $P(x,z)$, αντίστοιχα. Από την Πρόταση 6.1.3 το ερώτημα $\mathcal{Q}' = Q(x') \leftarrow P(x',y') \wedge A(y') \wedge R(x',z') \wedge A(z')$ μπορεί να παραχθεί με εκτύλιξη στο $\mathcal{Q}$. Για $\theta = \{y \mapsto z', z \mapsto y'\}$ είναι φανερό ότι $\mathcal{Q}\theta \subseteq \mathcal{Q}'$, όμως, $y, z \in \mathsf{bound}(Q)$ και έτσι το $\mathcal{Q}'$ δεν ικανοποιεί τις συνθήκες της Πρότασης 6.1.7. $\diamondsuit$

Παρόλα αυτά, σημειώνουμε ότι σε όλα τα προηγούμενα παραδείγματα, δύο διαφορετικά σύνολα εκτύλιξης περιλαμβάνουν ένα άτομο με το ίδιο όνομα κατηγορήματος (για παράδειγμα το E στο Παράδειγμα 6.1.6 και τα R και P στο Παράδειγμα 6.1.8). Αναμένουμε ότι τέτοιες περιπτώσεις εμφανίζονται σπάνια, και ότι για τα άτομα $A_1, \ldots, A_n$ σε ν διαφορετικά σύνολα εκτύλιξης δε θα υπάρχει καποιο A_j το οποίο να χρησιμοποιεί το ίδιο όνομα κατηγορήματος με κάποιο άλλο άτομο σε διαφορετικό σύνολο εκτύλιξης. Τότε, μπορούμε να συμπεράνουμε ότι το ερώτημα με σώμα $A_1 \wedge \ldots \wedge A_n$ δεν είναι περιττό. Ο πρακτικός μας αλγόριθμος εφαρμόζει παρόμοιες στρατηγικές, προκειμένου να εντοπιστούν τα ερωτήματα που δεν υπάγονται σε άλλα. Ο άμεσος εντοπισμός μη περιττών ερωτημάτων, μπορεί να αντικαταστήσει εν μέρη, κάποιο τελικό βήμα ελέγχου υπαγωγής ερωτημάτων, με αποτέλεσμα ο αλγόριθμος επαναγραφής να αποκρίνεται ταχύτερα [132].

Κεφάλαιο 7

Πειραματική αξιολόγηση

Υλοποιήσαμε τους αλγόριθμους επαναγραφής που σχεδιάσαμε για την DL-Lite, την $\mathcal{ELHI}$, και τη Horn-$\mathcal{SHIQ}$, λαμβάνοντας υπόψη και τις βελτιστοποιήσεις που περιγράψαμε στο Κεφάλαιο 6, στο πρωτότυπο σύστημα Rapid [40, 128, 129][1]. Το σύστημα Rapid κατασκευάζει την επαναγραφή ενός συζευκτικού ερωτήματος $\mathcal{Q}$, και μίας οντολογίας εισόδου. Η έξοδος του συστήματος είναι ένα datalog προγράμμα, ενώ στην περίπτωση των οντολογιών σε DL-Lite, η επαναγραφή είναι δυνατό να έχει τη μορφή ΣΣΕ, αν το επιλέξει ο χρήστης (βλ. Κεφάλαιο 6).

Για τους σκοπούς της εργασίας, επεκτείναμε σημαντικά τα υπάρχοντα σύνολα αξιολόγησης των συστημάτων επαναγραφής, συμπεριλαμβάνοντας ενδιαφέρουσες οντολογίες μεγάλης κλίμακας, που χρησιμοποιούνται σε εφαρμογές του πραγματικού κόσμου. Στις επόμενες ενότητες παρουσιάζουμε τα πειραματικά αποτέλεσματα της αξιολόγησης για οντολογίες εισόδου που είναι διατυπωμένες στις γλώσσες ΠΛ DL-Lite, $\mathcal{ELHI}$, και Horn-$\mathcal{SHIQ}$.

7.1 Πειραματικά αποτελέσματα στην DL-Lite

Στην περίπτωση της DL-Lite, συγκρίναμε το Rapid με τα συστήματα Requiem [110], Presto [116], και Clipper [45]. Χρησιμοποιοήσαμε DL-Lite εκδοχές των οντολογιών OBO Protein,[2] NCI 3.12e,[3] και OpenGALEN2[4]. Ο Πίνακας 7.4 παρέχει τα σχετικά στατιστικά για αυτές τις οντολογίες. Επιπρόσθετα, χρησιμοποιήσαμε όλες τις DL-Lite οντολογίες που προτάθηκαν στο [108].

Τα πειραματικά αποτελέσματα αφορούν το χρόνο υπολογισμού της επαναγραφής και το μέγεθός της. Στους σχετικούς πίνακες, όπως και σε όσους παρουσιάζονται στις επόμενες ενότητες, οι χρόνοι εκφράζονται με ακρίβεια χιλιοστών του δευτερολέπτου. Σημειώνουμε ότι η ένδειξη "—" δηλώνει ότι το αντίστοιχο σύστημα υπερέβει το χρόνο όριο τερματισμού, που θέσαμε τις 3 ώρες.

Στον Πίνακα 7.2 παρουσιάζουμε τα αποτελέσματα για τις ενδιαφέρουσες περιπτώσεις από τις οντολογίες που προτάθηκαν στο [108], που είναι σχετικά μικρές και

[1] http://www.image.ece.ntua.gr/~achort/rapid/
[2] http://www.obofoundry.org
[3] http://evs.nci.nih.gov/ftp1/NCI_Thesaurus
[4] http://www.opengalen.org

Πίνακας 7.1: Στατιστικά για τις DL-Lite οντολογίες μεγάλης κλίμακας

$\mathcal{T}$	♯αξιώματα
DL-Lite	
OBO Protein	43351
NCI	53341
OpenGALEN2	499228

απλές. Λόγω της απλής δομής των οντολογιών, επιλέγουμε η έξοδος των συστημάτων να έχει τη μορφή ΣΣΕ. Ο σκοπός της συγκεκριμένης αξιολόγησης είναι η παρουσίαση της επίδοσης του Rapid, ως προς τις βελτιστοποιήσεις που περιγράψαμε στο Κεφάλαιο 6. Σύμφωνα με τα αποτελέσματα, το Rapid έχει παρόμοια επίδοση με τα υπόλοιπα συστήματα σε ορισμένες περιπτώσεις, ενώ σε άλλες αποκρίνεται πολύ γρηγορότερα. Για παράδειγμα, για το ερώτημα 5 επί της AX οντολογίας, το σύστημά μας υπολογίζει την επαναγραφή σε 1 δευτερόλεπτο, ενώ το Requiem και το Presto χρειάζονται περίπου 1 ώρα. Από την ανάλυση των αποτελεσμάτων, προκύπτει ότι το Requiem αφιερώνει τον περισσότερο χρόνο σε ελέγχους υπαγωγής επί της παραχθείσας επαναγραφής, τους οποίους το Rapid καταφέρνει να αποφύγει λόγω της προσέγγισης που περιγράφεται στην Πρόταση 6.1.7.

Πίνακας 7.2: Αξιολόγηση για τις οντολογίες του Requiem

	Χρόνος (hh:mm:ss.msec)			Μέγεθος επαναγραφής		
Ø	Rapid	Requiem	Presto	Rapid	Requiem	Presto
P5X	0.004	0.012	0.026	14	14	14
	0.021	0.137	0.103	25	25	26
	0.022	0.313	0.203	58	58	37
	0.040	4.160	2.005	179	179	82
	0.445	2:17.683	51.538	718	718	251
UX	0.002	0.023	0.413	5	5	5
	0.010	0.170	0.047	1	1	1
	0.013	0.691	0.049	12	12	12
	0.003	10.317	0.043	5	5	5
	0.005	34.232	0.052	25	25	25
AX	0.016	0.028	1.856	41	41	41
	0.158	1.595	4.108	1431	1431	1431
	0.108	16.751	1:00.466	4466	4466	4466
	0.389	13.068	33.729	3159	3159	3159
	0.964	>1h	>1h 21m	32921	32921	36330

Στον Πίνακα 7.3 παρουσιάζουμε τα αποτελέσματα για τις οντολογίες μεγάλης κλίμακας σε DL-Lite. Για κάθε μία από αυτές κατασκευάσαμε 5 ερωτήματα. Δεδομένου ότι οι οντολογίες είναι αρκετά μεγάλες, επιλέξαμε η έξοδος όλων των συστημάτων να είναι στη μορφή datalog προγράμματος. Όπως γίνεται κατανοητό, το πλήθος των ερωτημάτων στην περίπτωση μιας ΣΣΕ επαναγραφής, θα ήταν απαγορευτικό λόγω του μεγάλου μεγέθους των οντολογιών εισόδου. Αρχικά, παρατηρούμε ότι το Presto και το Clipper απέτυχαν να υπολογίσουν επαναγραφή για την OpenGALEN2 στο προκαθορισμένο χρόνο, ενώ το Requiem χρειάστηκε εως 6 λεπτά, και το Rapid μερικά χιλιοστά του δευτερολέπτου. Παρόμοιες παρατηρήσεις μπορούν να γίνουν και

για τις δύο άλλες οντολογίες. Αξιοσημείωτες περιπτώσεις είναι όλα τα ερωτήματα επί της NCI για το Clipper, που αποκρίθηκε σε 12-14 λεπτά, για κάθε ένα από αυτά, το ερώτημα 5 επί της OBO Protein, και το ερώτημα 3 επί της NCI για το Requiem, για το οποίο δεν κατάφερε να υπολογίσει επαναγραφή, και όλα τα ερωτήματα επί της OBO Protein και της NCI για το Presto, καθώς χρειάστηκε περίπου 1 και 2 ώρες, αντίστοιχα μέχρι να τερματίσει. Το Rapid ήταν πιο αργό σύστημα σε σύγκριση με το Clipper, μόνο στο ερώτημα 5, επί της OBO Protein, για το οποίο χρειάστηκε 5 λεπτά και 26 δευτερόλεπτα. Η ανάλυση της συγκεκριμένης περίπτωσης έδειξε ότι ο περισσότερος χρόνος αφιερώθηκε σε από-κάτω-προς-τα-πάνω ελέγχους υπαγωγής που εξασφαλίζουν ένα πιο συμπαγές datalog πρόγραμμα, χωρίς περιττά ερωτήματα που υπάγονται σε άλλα. Ο πραγματικός χρόνος υπολογισμού της επαναγραφής, που προκύπτει εαν δεν συνυπολογίσουμε το χρόνο που αφιερώθηκε στον εντοπισμό ερωτημάτων που υπάγονται σε άλλα, είναι μόνο μερικά δευτερόλεπτα.

Σημειώνουμε ότι κανένα άλλο σύστημα δεν πραγματοποιεί από-κάτω-προς-τα-πάνω έλεγχο υπαγωγής και έτσι, στο ερώτημα 5 επί της OBO Protein, το Clipper υπολογίζει μία επαναγραφή που έχει το διπλάσιο μέγεθος από εκείνη που υπολογίζεται από το Rapid, καθώς περιλαμβάνει πολλές προτάσεις που είναι ισοδύναμες αν λάβουμε υπόψη μετονομασίες μεταβλητών. Το Presto κατάφερε να υπολογίσει μία επαναγραφή με μόλις 2670 προτάσεις που είναι ένας αρκετά μικρός αριθμός σε σύγκριση με την επαναγραφή που υπολογίστηκε για προηγούμενα ερωτήματα. Σχετικά με το μέγεθος της επαναγραφής σε άλλες περιπτώσεις, όλα τα συστήματα κατασκεύασαν περίπου τον ίδιο αριθμό προτάσεων, εκτός από το ερώτημα 5 επί της OBO Protein, όπως αναφέραμε παραπάνω, και τα ερωτήματα 1, 4, 5 επί της NCI για το Requiem. Όπως γίνεται κατανοητό, οι μικρές διαφορές στο μεγέθος της επαναγραφής που επιστρέφει το κάθε σύστημα, μπορούν να αποδοθούν στη διαφορετική μορφή του datalog προγράμματος, καθώς και σε βελτιστοποιήσεις που έχουν σα στόχο να ανιχνεύουν και να αφαιρούν τις περιττές προτάσεις που υπάγονται σε άλλες.

Πίνακας 7.3: Αξιολόγηση με DL-Lite οντολογίες

	Χρόνος (hh:mm:ss.msec)			
$\mathcal{T}$	Rapid	Requiem	Presto	Clipper
OBO Protein	0.154	4.781	59:09.975	1:04.782
	1.160	45.473	1:04:36.706	1:07.770
	7.264	9:51.364	1:17:22.561	1:07.084
	10.613	12:31.979	59:35.577	1:03.496
	5:26.680	–	1:09:05.760	1:08.245
NCI	0.057	4.423	1:58:34.415	11:49.193
	0.043	36.895	2:02:14.930	12:00.461
	0.154	–	2:02:32.860	13:16.689
	0.063	1:17:51.520	2:04:19.429	13:55.669
	0.041	9:39.122	2:09:21.690	13:27.232
OpenGALEN2	0.001	0.024	–	–
	0.099	6:04.373	–	–
	0.001	0.111	–	–
	0.004	5:46.020	–	–
	0.001	0.022	–	–

	Μέγεθος επαναγραφής			
$\mathcal{T}$	Rapid	Requiem	Presto	Clipper
OBO Protein	29	27	48	29
	1356	1356	2621	1356
	33919	33887	33888	33919
	34879	34733	35416	34879
	27907	–	2670	54430
NCI	488	5002	469	488
	1804	1765	1766	1804
	4143	–	3546	4143
	1875	219150	1917	1875
	256	64500	208	340
OpenGALEN2	3	2	–	–
	1276	1152	–	–
	18	16	–	–
	155	147	–	–
	1	1	–	–

7.2 Πειραματικά αποτελέσματα στην $\mathcal{ELHI}$

Στην περίπτωση της $\mathcal{ELHI}$, συγκρίναμε το Rapid και πάλι, με τα συστήματα Requiem [110], Presto [116], και Clipper [45]. Η αξιολόγηση των συστημάτων επαναγραφής έγινε με τις $\mathcal{ELHI}$ εκδοχές των LUBM,[5] OBO Protein, NASA SWEET 2.3,[6] PERIODIC Table,[7] NotGALEN,[8] GALEN-Doctored [61], της OpenGALEN2, και της ExtendedDNS[9]. Ο Πίνακας 7.4 παρέχει τα σχετικά στατιστικά για αυτές τις οντολογίες.

Πίνακας 7.4: Στατιστικά για τις $\mathcal{ELHI}$ οντολογίες μεγάλης κλίμακας

$\mathcal{T}$	♯αξιώματα
$\mathcal{ELHI}$	
OBO Protein	52383
NASA SWEET	6415
PERIODIC Table	9579
NotGALEN	10967
GALEN-Doctored	8556
OpenGALEN2	64508
ExtendedDNS	753

Στον Πίνακα 7.6 παρουσιάζονται τα πειραματικά αποτελέσματα για την $\mathcal{ELHI}$. Όπως προκύπτει, το Rapid αποκρίνεται ταχύτερα για τις οντολογίες μεσαίου μεγέθους, ενώ η απόδοσή του ξεπερνά εκείνη των υπόλοιπων συστημάτων, σε περιπτώσεις μεγαλύτερων και περισσότερο σύνθετων οντολογιών. Μάλιστα, σε αρκετές περιπτώσεις τα υπόλοιπα συστήματα δεν τερμάτισαν μέσα στο προκαθορισμένο χρονικό όριο. Κατ' εξαίρεση, στο ερώτημα 5 επί της OBO Protein, το Rapid είχε χειρότερη επίδοση από ότι το Clipper, γεγονός που οφείλεται στον τελικό, από-κάτω-προς-τα-πάνω έλεγχο υπαγωγής προτάσεων. Το σύστημα Rapid φαίνεται πως σε αυτή την περίπτωση, αφιερώνει αρκετό χρόνο για τον εντοπισμό προτάσεων που υπάγονται σε άλλες, με σκοπό να μην τις συμπεριλάβει στην datalog επαναγραφή. Πράγματι, όπως επιβεβαιώνεται, το Clipper επιστρέφει ένα πρόγραμα διπλάσιου μεγέθους από ότι το Rapid. Αξιοσημείωτη είναι και η περίπτωση της ExtendedDNS οντολογίας όπου, παρόλο που δεν πρόκειται για μία οντολογία πολύ μεγάλου μεγέθους, και τα δύο συστήματα Requiem και Clipper αποτυγχάνουν να τερματίσουν σε αντίθεση με το Rapid.

Στο σημείο αυτό, διατυπώνουμε ορισμένες ενδιαφέρουσες παρατηρήσεις που στηρίζονται στις ποικίλες εκδοχές της GALEN οντολογίας που χρησιμοποιήσαμε. Οι οντολογίες NotGALEN και GALEN-Doctored βασίζονται σε δύο πρώιμες εκδοχές της GALEN που έχουν μία σχετικά απλή δομή. Τα αποτελέσματα δείχουν ότι το Rapid παρουσιάζει καλή επίδοση κατασκευάζοντας τις επαναγραφές σε λίγα δευτερόλεπτα. Το Clipper απαιτεί περίπου 25 λεπτά στην περίπτωση της NotGALEN, που

[5] http://swat.cse.lehigh.edu/projects/lubm
[6] http://sweet.jpl.nasa.gov/ontology
[7] http://www.cs.man.ac.uk/~stevensr/ontology
[8] http://www.cs.ox.ac.uk/isg/ontologies/lib/GALEN/not-galen
[9] http://www.loa.istc.cnr.it

Πίνακας 7.5: Αξιολόγηση για την $\mathcal{ELHI}$ με τα ερωτήματα από το SyGENiA

	Time (hh:mm:ss.msec)			Rewriting size		
Ø	Rapid	Req.	Clip.	Rapid	Req.	Clip.
NASA	0.027	19.776	–	510.90	3818.74	–
ExtDNS	0.065	–	–	1613.91	–	–

είναι η απλούστερη από τις δύο οντολογίες, ενώ δεν τερματίζει στο χρονικό όριο, για την GALEN Doctored. Το σύστημα Requiem δεν αποκρίνεται για καμία. Τα αποτελέσματα είναι διαφορετικά για την OpenGALEN2, που αποτελεί μία οντολογία πρόκληση για όλα τα συστήματα λόγω της ιδιαίτερα σύνθετης δομής της. Πράγματι, το Requiem και το Clipper δεν κατάφεραν να τερματίσουν για κανένα από τα ερωτήματα επί της OpenGALEN2. Το Rapid υπολογίζει την επαναγραφή για το ερώτημα 1, ενώ για το ερώτημα 2 τερματίζει σε περίπου, 3 ώρες. Για τα ερωτήματα 3, 4, 5 το Rapid δεν κατάφερε να τερματίσει. Η μελέτη της συσμπεριφοράς του συστήματος σε αυτές τις περιπτώσεις, έδειξε ότι η κακή επίδοση οφείλεται σε επεκταμένη εφαρμογή του κανόνα συνάρτησης, γεγονός που οδηγεί σε αύξηση των προτάσεων που συμμετέχουν ως δευτερεύουσες στον κανόνα της συρρίκνωσης.

Μία ακόμα ενδιαφέρουσα περίπτωση είναι το ερώτημα 1 για όλες τις διαφορετικές εκδοχές της GALEN. Το ερώτημα αυτό παρά την απλή του μορφή, $Q(x) \leftarrow \mathsf{Heme}$, περιλαμβάνει την έννοια Heme που βρίσκεται χαμηλά στην ιεραρχία των κλάσεων της οντολογίας. Μάλιστα, η έννοια σχετίζεται με ένα απομονωμένο τμήμα της οντολογίας GALEN. Για το λόγο αυτό η επαναγραφή του ερωτήματος περιλαμβάνει λίγες μόνο προτάσεις. Σημειώνουμε ότι το Rapid και το Requiem μπορούν να αναγνωρίσουν αυτήν την ιδιομορφία, και να αποκριθούν αμέσως, περιορίζοντας την εφαρμογή των κανόνων της επίλυσης μόνο στις σχετικές έννοιες που αφορούν μικρό τμήμα της οντολογίας. Αντίθετα, το Clipper επεξεργάζεται όλη την οντολογία προκειμένου να υπολογίσει την επαναγραφή του ερωτήματος.

Τέλος, αξιολογήσαμε την επίδοση των συστημάτων με ερωτήματα που κατασκευάστηκαν με τη βοήθεια του εργαλείου που παρουσιάζεται στο [73]. Ο Πίνακας 7.5 παρουσιάζει τα σχετικά αποτελέσματα, όπου φαίνονται οι μέσοι χρόνοι και η μέση τιμή του μεγέθους των επαναγραφών για την κάθε οντολογία, λόγω του ιδιαίτερα μεγάλου πλήθους ερωτημάτων. Από τη μελέτη των αποτελεσμάτων, είναι φανερό ότι το Rapid παρουσιάζει σημαντικά καλύτερη επίδοση σε σύγκριση με τα Requiem και Clipper.

Συνοψίζοντας, τα πειραματικά αποτελέσματα καταδεικνύουν ότι το Rapid επιτυγχάνει ιδιαίτερα βελτιωμένους χρόνους απόκρισης για τις σύνθετες και μεγάλες οντολογίες, ενώ δεν καταφέρνει να διαχειριστεί ορισμένες ιδιαίτερες περιπτώσεις οντολογιών σε $\mathcal{ELHI}$ που θεωρούνται εξαιρετικά απαιτητικές από όλα τα σύγχρονα συστήματα συλλογιστικής της OWL. Σημειώνουμε ότι ο υπολογισμός της επαναγραφής οντολογιών μεγάλης κλίμακας αποτελεί πρόκληση για όλα τα σύγχρονα συστήματα.

Πίνακας 7.6: Αξιολόγηση για τις $\mathcal{ELHI}$ οντολογίες

	Χρόνος (hh:mm:ss.msec)			Μέγεθος επαναγραφής		
Ø	Rapid	Requiem	Clipper	Rapid	Requiem	Clipper
OBO Protein	9.626	–	1:42.249	51641	–	51641
	29.764	–	1:45.742	52877	–	52877
	6.247	–	1:46.322	51614	–	51614
	16.940	–	1:44.356	52407	–	52407
	11:23.172	–	1:49.475	79427	–	105950
NASA SWEET	0.095	0.410	–	170	288	–
	0.015	0.840	–	507	1800	–
	0.013	0.753	–	660	1945	–
	0.028	2.551	–	1097	3380	–
	0.031	42.703	–	1107	19515	–
PERIODIC Table	0.147	3:31.069	20.512	1103	6800	2892
	0.035	4:11.178	19.857	879	6941	2892
	0.064	4:26.803	19.610	1653	6889	2892
	0.054	4:35.427	20.817	1609	8077	2849
	0.094	10:30.913	20.936	1743	57054	2893
NotGALEN	0.002	0.006	23:57.628	1	1	1
	1.527	–	24:35.363	11907	–	11756
	0.972	–	22:50.269	11913	–	11769
	0.001	–	22:28.542	11	–	11
	0.988	–	22:59.788	11913	–	11760
GALEN-Doctored	0.004	0.009	–	3	2	–
	1.772	–	–	9563	–	–
	0.772	–	–	9566	–	–
	0.001	–	–	11	–	–
	0.931	–	–	9576	–	–
OpenGALEN2	0.002	0.01	–	3	2	–
	2:49:21.804	–	–	160817	–	–
	–	–	–	–	–	–
	–	–	–	–	–	–
	–	–	–	–	–	–
ExtendedDNS	0.272	–	–	1996	–	–
	0.085	–	–	1999	–	–
	0.002	–	–	96	–	–
	0.009	–	–	281	–	–
	0.077	–	–	2042	–	–

7.3 Πειραματικά αποτελέσματα στη Horn-$\mathcal{SHIQ}$

Στην περίπτωση της Horn-$\mathcal{SHIQ}$ συγκρίναμε το Rapid μόνο με το σύστημα Clipper, καθώς αποτελεί το μοναδικό διαθέσιμο σύστημα επαναγραφής συζευκτικού ερωτήματος σε Horn-$\mathcal{SHIQ}$ οντολογίες. Κατά την αξιολόγηση των συστημάτων χρησιμοποιήθηκαν τα Horn-$\mathcal{SHIQ}$ υποσύνολα των οντολογιών NASA SWEET 2.3,[10] Periodic,[11] και DOLCE2.1Lite-Plus[12]. Επιπλέον, πέρα από τις προαναφερθείσες οντολογίες μεγάλης κλίμακας, χρησιμοποιήσαμε την UOBM οντολογία, που περιέχεται στο σύνολο αξιολόγησης του Clipper, καθώς και την Bio-PAX οντολογία [13]. Τα στατιστικά των οντολογιών παρουσιάζονται στον Πίνακα 7.7.

Πίνακας 7.7: Στατιστικά για τις Horn-$\mathcal{SHIQ}$ οντολογίες

$\mathcal{T}$	♯Αξιώματα	$\mathcal{T}$	♯Αξιώματα
UOBM	207	Periodic_full	43689
NASA SWEET	11578	DOLCE2.1	1055
BioPAX	333		

Στον Πίνακα 7.8 παρουσιάζουμε τα αποτελέσματα για τις οντολογίες μικρής κλίμακας, UOBM, και BioPAX. Για την UOBM χρησιμοποιήσαμε τα 10 ερωτήματα που περιλαμβάνει το σύνολο αξιολόγησης του Clipper, ενώ για την πρίπτωση της Bio-PAX κατασκευάσαμε 10 ερωτήματα. Σαν γενική παρατήρηση, σημειώνουμε ότι το Rapid αποκρίνεται ταχύτερα από ότι το σύστημα Clipper. Το πλήθος των προτάσεων της επαναγραφής είναι σχεδόν τα ίδιο και για τα δύο συστήματα, ενώ μικρές διαφορές οφείλονται στη διαφορετική δομή του datalog προγράμματος. Για παράδειγμα, στην περίπτωση ενός ερωτήματος της μορφής $Q(x) \leftarrow A(x) \wedge B(x)$ και των προτάσεων $A(x) \leftarrow C_i(x)$, με $1 \leq i \leq n$, το Clipper θα συμπεριλάβει στην επαναγραφή όλες τις προτάσεις της μορφής $Q(x) \leftarrow C_i(x) \wedge B(x)$. Χαρακτηριστική περίπτωση αποτελεί το ερώτημα 3.

Στον Πίνακα 7.9 παρουσιάζουμε τα πειραματικά αποτελέσματα για την περίπτωση των οντολογιών μεγαλύτερου μεγέθους. Για κάθε μία από αυτές κατασκευάσαμε 5 ερωτήματα. Παρατηρούμε ότι το σύστημα Rapid αποκρίνεται σε λίγα μόνο δευτερόλεπτ, ενώ το σύστημα Clipper δεν κατάφερε να τερματίσει στο προκαθορισμένο όριο χρόνου.

Τέλος, αξιοποιήσαμε και πάλι το εργαλείο που παρουσιάζεται στο [73] προκειμένου να κατασκευάσουμε επιπρόσθετα ερωτήματα για τις οντολογίες μεγάλης κλίμακας, καθώς και τη μικρότερη οντολογία BioPAX. Στις οντολογίες μεγάλης κλίμακας το Rapid τερματίζει με μερικά δευτερόλεπτα, συγκεκριμένα για την NASA SWEET τερματίζει κατά μέσο όρο σε 7.53 χιλιοστά του δευτερολέπτου, για την Periodic_full σε 65.63 χιλιοστά του δευτερολέπτου, και για την DOLCE2.1 σε 256.07 χιλιοστά του δευτερολέπτου. Το Clipper δεν αποκρίνεται σε κανένα ερώτημα. Στην περίπτωση της BioPAX κατασκευάστηκαν 218 ερωτήματα, για τα οποία και τα δύο συστήματα

[10] https://sweet.jpl.nasa.gov/
[11] http://www.cs.man.ac.uk/ stevensr/ontology
[12] http://www.loa.istc.cnr.i-ld/DOLCE.html
[13] http://biopax.org

Πίνακας 7.8: Αξιολόγηση για τις Horn-$\mathcal{SHIQ}$ οντολογίες

UOBM (207 axioms)				Bio-PAX (333 axioms)			
$\mathcal{I}_{\mathrm{RHS}}$		CLIPPER		$\mathcal{I}_{\mathrm{RHS}}$		CLIPPER	
t	Rew.	t	Rew.	t	Rew.	t	Rew.
11	3	64	2	53	25	38	25
18	14	56	16	3	8	38	9
65	109	1415	920	1	5	44	5
25	22	59	45	18	25	46	25
18	16	67	33	2	7	37	7
20	13	117	16	2	7	38	7
21	14	65	15	30	87	37	87
14	10	61	25	2	8	42	8
28	31	55	33	7	29	40	29
23	23	61	23	1	2	54	3

Πίνακας 7.9: Αξιολόγηση για τις Horn-$\mathcal{SHIQ}$ οντολογίες μεγάλης κλίμακας

	Axioms	t	Rew. size
NASA SWEET	11578	56	170
		142	399
		177	551
		414	979
		348	989
Periodic_full	43689	29	23
		1768	533
		1336	631
		129	195
		605	714
DOLCE2.1	1055	1314	1192
		1230	1194
		35	85
		163	205
		1379	1192

αποκρίθηκαν σε μερικά δευτερόλεπτα, με το Rapid να τερματίζει ταχύτερα, σχεδόν στο ένα τρίτο του χρόνου που χρειάστηκε το Clipper.

Καταλήγοντας, αξίζει να σημειώσουμε ότι ο υπολογισμός του datalog προγράμματος σε εύλογο χρόνο αποτελεί μία από τις βασικές απαιτήσεις των συστημάτων απάντησεις ερωτημάτων σε οντολογίες. Όπως γίνεται φανερό από τα πειραματικά αποτελέσματα, το σύστημα επαναγραφής Rapid αποκρίνεται σε ιδιαίτερα ανταγωνιστικούς χρόνους σε σύγκριση με τα διαθέσιμα συστήματα επαναγραφής.

Μέρος IV

Επίλογος

Κεφάλαιο 8

Σχετική βιβλιογραφία και συνεισφορά εργασίας

Η απάντηση των συζευκτικών ερωτημάτων που διατυπώνονται σε κάποια βάση γνώσης αποτελεί πλέον, ένα σύνηθες πρόβλημα για τις σύγχρονες εφαρμογές που χρησιμοποιούν οντολογίες προκειμένου να περιγράψουν το σημασιολογικό περιεχόμενο των δεδομένων τους. Η απάντηση ερωτημάτων με τη διαδικασία της επαναγραφής έχει μελετηθεί θεωρητικά για γλώσσες περιγραφικής λογικής διαφορετικής εκφραστικότητας, που αντιστοιχίζονται στα διαφορετικά υποσύνολα της γλώσσας OWL. Η θεωρητική μελέτη του προβλήματος έχει καρποφορήσει σημαντικά αποτελέσματα που αφορούν τη δυνατότητα παραγωγής της επαναγραφής και χαρακτηρίζουν την πολυπλοκότητα της διαδικασίας [32, 78, 57, 23, 77]. Παράλληλα, το πρόβλημα έχει επίσης μελετηθεί από μία περισσότερο πρακτική σκοπία, με αποτέλεσμα το σχεδιασμό και την ανάπτυξη συστημάτων επαναγραφής [33, 110, 116, 102, 40, 113, 45, 132].

Η μέθοδος της επαναγραφής αποτελεί ιδιαίτερα δημοφιλή προσέγγιση για την απάντηση συζευκτικών ερωτημάτων με βάση οντολογίες. Βασικό χαρακτηριστικό της προσέγγισης είναι ότι η επαναγραφή, είτε με τη μορφή datalog προγράμματος, είτε με τη μορφή συνόλου συζευκτικών ερωτημάτων, μπορεί να αξιοποιηθεί για την ανάκτηση των απαντήσεων του συζευκτικού ερωτήματος από συστήματα και τεχνολογίες του σημασιολογικού ιστού και των βάσεων δεδομένων. Η μέθοδος είναι ιδιαίτερα δημοφιλής, ειδικά στις περιπτώσεις οντολογιών που διατυπώνονται σε γλώσσες χαμηλής εκφραστικότητας, όπως είναι η DL-Lite. Καθώς αυξάνεται η εκφραστικότητα της γλώσσας αναπαράστασης της οντολογίας, η τεχνική της επίλυσης μπορεί να διευκολύνει σημαντικά το σχεδιασμό του αλγορίθμου επαναγραφής. Ένα βασικό πλεονέκτημα της επίλυσης είναι η δυνατότητα επέκτασής της σε υψηλότερες εκφραστικότητες. Επιπλέον, η μέθοδος παρέχει τη δυνατότητα υιοθέτησης πληθώρας τεχνικών βελτιστοποίησης που έχουν προκύψει από τη μακροχρόνια μελέτη της επίλυσης στη λογική πρώτης τάξης.

Η πρώτη εργασία που αφορά την παραγωγή επαναγραφής με βάση την επίλυση, παρουσιάζεται στο [99]. Σε αυτή την εργασία περιγράφεται ένας αλγόριθμος επαναγραφής που υλοποιήθηκε στο σύστημα KAON2. Ο αλγόριθμος εφαρμόζει τους κανόνες της βασικής υπέρθεσης σε $\mathcal{SHIQ}$ οντολογίες, με σκοπό να κατασκευάσει ένα datalog πρόγραμμα το οποίο συνεπάγεται τα ίδια βασικά γεγονότα με την οντολο-

γία. Ο αλγόριθμος υποστηρίζει συζευκτικά ερωτήματα χωρίς μεταβλητές απάντησης και επομένως, δεν είναι κατάλληλος για την επαναγραφή συζευκτικών ερωτημάτων.

Στην εργασία [110] προτάθηκε ο αλγόριθμος που υλοποιείται στο σύστημα Requiem. Όπως είναι γνωστό, ο αλγόριθμος εφαρμόζει εξαντλητικά τον κανόνα της δυαδικής επίλυσης με σκοπό την παραγωγή της επαναγραφής συζευκτικών ερωτημάτων που διατυπώνονται σε DL-Lite και $\mathcal{ELHI}$ οντολογίες.

Ο αλγόριθμος που προτάθηκε στο [102] βασίζεται επίσης, στη μέθοδο της SLD επίλυσης και αφορά οντολογίες που είναι διατυπωμένες στη γραμμική Datalog$^{\pm}$. Ο αλγόριθμος υλοποιήθηκε στο σύστημα Nyaya.

Οι παραπάνω αλγόριθμοι επαναγραφής που εφαρμόζουν τη μέθοδο της επίλυσης, δεν καταφέρνουν να αποφύγουν το βασικό μειονέκτημα της μεθόδου, που σχετίζεται με τη διάδοση συναρτησιακών συμβόλων στις παραγόμενες προτάσεις, καθώς και την κατασκευή προτάσεων που δεν συπεριλαμβάνονται στο τελικό αποτέλεσμα της επαναγραφής [122]. Κοινό χαρακτηριστικό τους είναι η εξαντλητική εφαρμογή των κανόνων της επίλυσης, με μία από-πάνω-προς-τα-κάτω προσέγγιση. Η εν λόγω προσέγγιση είναι πολύ πιθανό να οδηγήσει στην παραγωγή προτάσεων με συναρτησιακούς όρους που χαρακτηρίζονται από ολονένα και μεγαλύτερο βάθος. Ακόμα, στην περίπτωση του συστήματος KAON2, όπου ο αλγόριθμος εφαρμόζει βελτιστοποιήσεις και περιορισμούς που αφορούν την ταξινόμηση των όρων και την επιλογή κατάλληλης συνάρτησης επιλογής, παράγονται προτάσεις με συναρτησιακά σύμβολα, που δε λαμβάνονται υπόψη κατά την κατασκευή της επαναγραφής, ούτε όμως, συνεισφέρουν στην παραγωγή προτάσεων που συμπεριλαμβάνονται σε αυτή. Επιπρόσθετα, δεν αποκλείεται η παραγωγή προτάσεων με βάθος μεγαλύτερο από ένα, δηλαδή με εμφωλευμένες συναρτήσεις. Η εργασία μας αποτελεί την πρώτη προσπάθεια σχεδίασης ενός βελτιστοποιημένου αλγορίθμου επαναγραφής που εφαρμόζει τους κανόνες της επίλυσης με ελεγχόμενο τρόπο, με μία από κάτω-προς-τα-πάνω προσέγγιση, και αφορά εκφραστικές Horn-λογικές με ισότητα. Ο λογισμός μας δεν κατασκευάζει ποτέ όρους με εμφωλευμένες συναρτήσεις στο σώμα της πρότασης. Την ίδια στιγμή, η συγχώνευση περισσότερων κανόνων σε έναν, έχει ως αποτέλεσμα οι ενδιάμεσες προτάσεις με συναρτησιακούς όρους στο σώμα τους, να μην εμφανίζονται στην ακολουθία παραγωγής.

Τέλος, ο αλγόριθμος Presto [116] υπολογίζει την επαναγραφή συζευκτικού ερωτήματος σε DL-Lite οντολογίες με τη μορφή μη αναδρομικού datalog προγράμματος, και εισάγει βελτιστοποιήσεις που στοχεύουν στην κατασκευή ενός συμπαγούς datalog προγράμματος, μικρού μεγέθους. Ο αλγόριθμος Presto δεν εφαρμόζει κανόνες δυαδικής ανάλυσης, αλλά υπολογίζει την επαναγραφή παρόμοια με τον αλγόριθμο στο [33]. Αναφερόμαστε στον αλγόριθμο ως σχετική βιβλιογραφία, καθώς η τεχνική που υιοθετεί για την απομάκρυνση των δεσμευμένων μεταβλητών, παρουσιάζει ομοιότητες με εκείνη που εφαρμόζει ο αλγόριθμος του Rapid, ώστε να αποτρέπει την παραγωγή συμπερασματικών προτάσεων με συναρτήσεις. Συγκεκριμένα, στο Presto η απομάκρυνση των ατόμων του ερωτήματος με δεσμευμένες μεταβλητές, εξαρτάται από τη συμμετοχή των σχετικών κατηγορημάτων σε αξιώματα της οντολογίας με υπαρξιακό τελεστή. Η τεχνική αυτή παρουσιάζει ομοιότητες με τη βασική ιδέα που διέπει τον κανόνα της συρρίκνωσης, κατά την εφαρμογή του οποίου μία δεσμευμένη

μεταβλητή αντικαθίσταται από όρο συνάρτησης, που τελικά δεν εμφανίζεται στη συμπερασματική πρόταση. Όμως, η συγκεκριμένη τεχνική βελτιστοποίησης του Presto περιορίζεται στην εκφραστικότητα της DL-Lite που υποστηρίζει ο αλγόριθμος. Μάλιστα, ενώ ο αλγόριθμος υποστηρίζει την εκφραστικότητα της DL-Lite, μετασχηματίζει τα αξιώματα με προσοντούχο υπαρξιακό τελεστή στο δεξί σκέλος, όπως συμβαίνει και στο [33], χρησιμοποιώντας βοηθητικούς ρόλους, με αποτέλεσμα η τελική οντολογία να μην περιλαμβάνει τέτοιου είδους αξιώματα. Όπως γίνεται κατανοητό, αυτή η κωδικοποίηση των αξιωμάτων καθιστά μη προφανή τον τρόπο επέκτασης του αλγορίθμου σε άλλες εκφραστικότητες, ενώ, την ίδια στιγμή, αυξάνει τον αριθμό των προτάσεων της οντολογίας, γεγονός που μπορεί να έχει επιπτώσεις στην επίδοση του αλγορίθμου επαναγραφής.

8.1 Συνεισφορά

Η επίλυση αποτελεί ένα σημαντικό μηχανισμό συλλογιστικής που έχει μελετηθεί ιδιαίτερα στη λογική πρώτης τάξης. Η εφαρμογή των κανόνων της επίλυσης για τον υπολογισμό της επαναγραφής μιας οντολογίας, επιτρέπει την υιοθέτηση πληθώρας τεχνικών βελτιστοποίησης που έχουν προταθεί για εκφραστικά υποσύνολα της λογικής πρώτης τάξης. Την ίδια στιγμή όμως, οι αλγόριθμοι επαναγραφής που βασίζονται στην επίλυση, και έχουν αναπτυχθεί εως σήμερα, δεν είναι πρακτικοί στις περιπτώσεις οντολογιών μεγάλης κλίμακας. Βασική αιτία είναι το γεγονός ότι οι αλγόριθμοι είναι γενικού σκοπού και δεν καταφέρνουν να ξεπεράσουν το βασικό μειονέκτημα της μεθόδου, που σχετίζεται με την παραγωγή περιττών προτάσεων με συναρτησιακά σύμβολα. Ο κύριος στόχος της εργασίας μας ήταν η ανάπτυξη ενός αλγορίθμου επαναγραφής ο οποίος αξιοποιεί την μέθοδο της επίλυσης ώστε να υποστηρίζει γλώσσες υψηλής εκφραστικότητας, ενώ παράλληλα, προσαρμόζει τη μέθοδο στα ιδιαίτερα χαρακτηριστικά της γλώσσας και του προβλήματος της επαναγραφής, με αποτέλεσμα να αποτρέπει την παραγωγή προτάσεων με συναρτήσεις.

Μελετήσαμε το πρόβλημα της επαναγραφής με βάση την επίλυση για διαφορετικές εκφραστικότητες γλωσσών περιγραφικής λογικής. Αρχικά, για την περίπτωση των DL-Lite και $\mathcal{ELHI}$, χρησιμοποιήσαμε τον αλγόριθμο του Requiem σαν σημείο αναφοράς, ώστε να ορίσουμε ένα νέο λογισμό που εφαρμόζει τους κανόνες της επίλυσης με ελεγχόμενο τρόπο ακολουθώντας μία SLD προσέγγιση. Στόχος της κάθε αλυσίδας παραγωγής είναι η κατασκευή ενός συζευκτικού ερωτήματος που θα είναι μέλος της επαναγραφής. Συγκεκριμένα, στην περίπτωση της DL-Lite, κάθε εφαρμογή κανόνα καταλήγει σε κάποιο ερώτημα που συμπεριλαμβάνεται στην έξοδο του αλγορίθμου. Στην περίπτωση της $\mathcal{ELHI}$, που χαρακτηρίζεται από προτάσεις με αναδρομή, γίνεται μία περιορισμένη παραγωγή προτάσεων που δεν περιλαμβάνονται στην έξοδο, και είναι όμως, απαραίτητες για την παραγωγή άλλων, που αποτελούν μέλη της επαναγραφής. Στη συνέχεια, μελετήσαμε την περίπτωση της γλώσσας Horn-$\mathcal{SHIQ}$ που περιλαμβάνει προτάσεις με ισότητα. Χρησιμοποιώντας ως σημείο αναφοράς τον αλγόριθμο που υλοποιείται στο KAON2, ορίσαμε ένα λογισμό ο οποίος εφαρμόζει τους κανόνες της βασικής υπέρθεσης με ελεγχόμενο τρόπο, λαμβάνοντας υπόψη τα

ιδιαίτερα χαρακτηριστικά της γλώσσας. Επεκτείναμε τον λογισμό της $\mathcal{ELHI}$, και ορίσαμε νέους κανόνες που δεν παράγουν προτάσεις με συναρτήσεις, παραμόνο όταν είναι απαραίτητες για την κατασκευή της επαναγραφής. Επιπλέον, οι κανόνες που εφαρμόζονται δεν παράγουν ποτέ συναρτησιακούς όρους με βάθος μεγαλύτερο από ένα, στο σώμα των προτάσεων. Πρόκειται για την πρώτη προσπάθεια εφαρμογής των κανόνων της επίλυσης με ελεγχόμενο τρόπο σε λογικές με ισότητα.

Συγκεκριμένα, η συνεισφορά της εργασίας μας περγράφεται από τα παρακάτω:

- Αρχικά, εξετάσαμε το πρόβλημα για τη γλώσσα DL-Lite που σχετίζεται άμεσα με το υποσύνολο OWL QL της OWL 2 [97]. Σχεδιάσαμε έναν αλγόριθμο επαναγραφής που βασίζεται στην επίλυση, ο οποίος κατασκευάζει έμμεσα προτάσεις με συναρτησιακά σύμβολα, μόνο όταν είναι απαραίτητες για την κατασκευή προτάσεων που πρέπει να συμπεριληφθούν στην επαναγραφή. Στην πράξη, οι προτάσεις με συναρτήσεις δεν παράγονται ποτέ, καθώς "κρύβονται" σαν ενδιάμεσα αποτελέσματα κανόνων υπερεπίλυσης, που συγχωνεύουν περισσότερες εφαρμογές κανόνων σε ένα βήμα. Ονομάζουμε τον κανόνα υπερεπίλυσης του λογισμού μας, κανόνα συρρίκνωσης. Ιδιαίτερο ενδιαφέρον παρουσιάζει η δομή και οι αλληλεπιδράσεις των αξιωμάτων της DL-Lite, που επιτρέπουν την υιοθέτηση σημαντικών βελτιστοποιήσεων στην υλοποιήση του κανόνα της συρρίκνωσης.

- Στη συνέχεια, μελετήσαμε το πρόβλημα για τη γλώσσα $\mathcal{ELHI}$, που σχετίζεται άμεσα με το υποσύνολο OWL 2 EL της OWL 2 [97]. Οι αλληλεπιδράσεις των αξιωμάτων της $\mathcal{ELHI}$ είναι πολύ περισσότερο σύνθετες από ότι στην περίπτωση της DL-Lite, γεγονός που προσδίδει ιδιαίτερο ενδιαφέρον στο εγχείρημά μας. Αξίζει να αναφέρουμε ότι το πρόβλημα υπαγωγής εννοιών στην $\mathcal{ELHI}$ είναι ExpTime ενώ για την DL-Lite είναι P. Παρόλα αυτά, με προσεκτική επίλυση των κανόνων του λογισμού μας για την DL-Lite, συμπεραίνουμε πως μπορούμε να εφαρμόσουμε τη βασική ιδέα που διέπει τον κανόνα της συρρίκνωσης και στην περίπτωση της $\mathcal{ELHI}$. Έτσι, ο κανόνας επεκτείνεται ώστε να επιτρέπει περισσότερες από δύο διαφορετικές δευτερεύουσες προκείμενες προτάσεις, ενώ εφαρμόζεται και στον νέο τύπο προτάσεων που υποστηρίζει η $\mathcal{ELHI}$. Παρατηρούμε με ιδιαίτερο ενδιαφέρον πως τα αποτελέσματα της DL-Lite, μπορούν να επεκταθούν στην περισσότερο εκφραστική $\mathcal{ELHI}$.

- Τέλος, ο λογισμός μας προσαρμόζεται στην περίπτωση της Horn-$\mathcal{SHIQ}$, που υποστηρίζει προτάσεις με ισότητα. Πρόκειται για τον πρώτο αλγόριθμο επαναγραφής με επίλυση που αφορά τη Horn-$\mathcal{SHIQ}$ και εισάγει πληθώρα βελτιστοποιήσεων. Όπως και στην περίπτωση της $\mathcal{ELHI}$, ο κανόνας της συρρίκνωσης επεκτείνεται ώστε αυτή τη φορά να εφαρμοστεί, στις προτάσεις με ισότητα. Επιπλέον, οι κανόνες της βασικής υπέρθεσης εφαρμόζονται με βέλτιστο τρόπο σύμφωνα με τις ιδιομορφίες των αξιωμάτων της γλώσσας, με αποτέλεσμα να κατασκευάζονται μόνο προτάσεις που είναι απαραίτητες και αποτελούν μέλη της επαναγραφής.

- Πέρα από το σχεδιασμό του βελτιστοποιημένου αλγορίθμου επαναγραφής με βάση την επίλυση, προτείναμε και μία σειρά από πρακτικές τεχνικές βελτιστο-

ποίησης, που αφορούν την υλοποίηση του αλγορίθμου και την κατασκευή μίας συμπαγούς datalog επαναγραφής που δεν περιλαμβάνει μεγάλο πλήθος προτάσεων.

- Υλοποιήσαμε τον αλγόριθμό μας στο σύστημα Rapid, και προτείναμε νέα σύνολα οντολογιών αξιολόγησης που περιλαμβάνουν οντολογίες μεγάλου μεγέθους. Πρόκειται για την πρώτη αξιολόγηση συστημάτων επαναγραφής με ενδιαφέρουσες οντολογίες του πραγματικού κόσμου. Τα πειραματικά αποτελέσματα καταδεικνύουν ότι ενώ στις περιπτώσεις μικρών οντολογιών τα υπάρχοντα συστήματα αξιολόγησης είναι αρκετά ανταγωνιστικά, στις περιπτώσεις των σύνθετων οντολογιών, το Rapid τερματίζει σε λίγα δευτερόλεπτα, όταν τα υπόλοιπα συχνά δεν καταφένουν να τερματίσουν πριν από τις δύο ώρες. Ιδιαίτερο ενδιαφέρον παρουσιάζεται στην περίπτωση της Horn-$\mathcal{SHIQ}$, όπου το Rapid είναι το μοναδικό διαθέσιμο σύστημα για την επαναγραφή συζευκτικού ερωτήματος.

8.2 Θέματα προς μελέτη

Μία επέκταση αυτής της εργασίας με ιδιαίτερο ενδιαφέρον, είναι η προσαρμογή του αλγορίθμου μας που βασίζεται στην επίλυση, ώστε πέρα από τον υπολογισμό της επαναγραφής, να υπολογίζει τις ταξονομίες μιας οντολογίας. Η *ταξινόμηση οντολογίας* (*ontology classification*) είναι το πρόβλημα συλλογιστικής που αφορά τον υπολογισμό των σχέσεων υπαγωγής εννοιών και ρόλων μιας οντολογίας. Από τη μελέτη του προβλήματος και τη σχετική έρευνα, έχουν προκύψει διαφορετικοί αλγόριθμοι ταξινόμησης [51, 121, 8] που εισάγουν τεχνικές βελτιστοποίησης, και υλοποιούνται σε πρακτικά συστήματα συλλογιστικής, όπως το [52] που βασίζεται σε μηχανισμούς tableaux. Η ταξινόμηση είναι μία ιδιαίτερα δημοφιλής υπηρεσία συλλογιστικής για τα σύγχρονα συστήματα του σημασιολογικού ιστού, και για το λόγο αυτό, θεωρούμε ότι η επέκταση του αλγορίθμου μας με διαδικασίες ταξινόμησης, αποτελεί αντικείμενο εργασίας με υψηλό ενδιαφέρον. Επιπλέον, η καλή συμπεριφορά της μεθόδου της επίλυσης όπως υλοποιείται στο σύστημα μας, που επιβεβαιώνεται από τα πειραματικά αποτελέσματα, αποτελεί ισχυρό κίνητρο για την προσαρμογή της μεθόδου ώστε να υπολογίζει τις ταξονομίες μίας οντολογίας.

Άμεση εξέλιξη του αλγορίθμου επαναγραφής για τη Horn-$\mathcal{SHIQ}$ είναι η προσαρμογή του, ώστε να υποστηρίζει την πλήρη εκφραστικότητα των Horn-$\mathcal{SHOIQ}$ και Horn-$\mathcal{SROIQ}$ [105, 62] που αποτελούν το υπόβαθρο της OWL. Οι γλώσσες αυτές είναι βατές, καθώς η πολυπλοκότητα του προβλήματος απάντησης συζευκτικών ερωτημάτων παραμένει πολυωνυμική ως προς τα δεδομένα. Στην περίπτωση της Horn-$\mathcal{SHOIQ}$ η πολυπλοκότητα για τον υπολογισμό της επαναγραφής παραμένει εκθετική, ενώ είναι διπλά εκθετική για την Horn-$\mathcal{SROIQ}$ [104]. Προκειμένου ο αλγόριθμος επαναγραφής να ανταποκρίνεται σε αυτές τις πολύ εκφραστικές γλώσσες, θα πρέπει να υποστηρίζει ιδιαίτερα χαρακτηριστικά ρόλων που, που παρουσιάζουν ενδιαφέρον στις περιπτώσεις των οντολογιών με ονοματικές έννοιες, (nominals, $\mathcal{O}$). Έτσι, πρέπει να ληφθούν υπόψη οι *ανακλαστικοί ρόλοι* (*reflexive roles*), οι *μη-ανακλαστικοί ρόλοι* (*irreflexive roles*), και οι *αντισυμμετρικοί ρόλοι* (*antisymmetric roles*). Για πα-

ράδειγμα, για έναν αντισυμμετρικό ρόλο R με $(a, b) \in R^I$, δεν μπορεί να ισχύει ότι $(b, a) \in R^I$. Εαν ο ρόλος R είναι μη ανακλαστικός, τότε $(a, a) \notin R^I$ για όλα τα a του πεδίου Δ^I. Τέλος, θα πρέπει να ληφθούν υπόψη η σύνθεση ρόλων, $R \circ P$, καθώς και οι ρόλοι που είναι ξένοι μεταξύ τους, *ξένοι ρόλοι* (*disjoint roles*).

Σημειώνουμε επίσης, ότι ο αλγόριθμος επαναγραφής που προτείνουμε στην παρούσα εργασία, είναι δυνατό να προσαρμοστεί ώστε να υποστηρίζει υποσύνολα των γλωσσών Datalog$^{\pm}$, δεδομένου ότι οι εν λόγω γλώσσες αντιστοιχίζονται και επεκτείνουν ελαφρώς την οικογένεια γλωσσών της DL-Lite. Οι γλώσσες Datalog$^{\pm}$ περιλαμβάνουν τις βατές γλώσσες *γραμμική (linear)* Datalog$^{\pm}$ [26], guarded Datalog$^{\pm}$ [26], sticky [29] και sticky-join Datalog$^{\pm}$ [30]. Η γραμμική Datalog$^{\pm}$ περιλαμβάνει προτάσεις με ένα ένα μόνο άτομο στο σώμα και αποτελεί υποσύνολο της γλώσσας guarded Datalog$^{\pm}$. Οι προτάσεις στην guarded Datalog$^{\pm}$ διαθέτουν πάντοτε κάποιο άτομο στο σώμα που περιλαμβάνει όλες τις μεταβλητές της κεφαλής. Η (sticky-)join Datalog$^{\pm}$ θέτει περιορισμούς στις πολλαπλές εμφανίσεις μεταβλητών στο σώμα μιας πρότασης.

Ένα ανοικτό θέμα προς μελέτη είναι η δυνατότητα παραγωγής της επαναγραφής σε περιπτώσεις που τα ερωτημάτα περιλαμβάνουν μεταβατικούς ρόλους. Στην περίπτωση αυτή, το ζητούμενο είναι να διερευνήσουμε εαν ο αλγόριθμος επαναγραφής παραμένει πλήρης και τερματίζει, εφαρμόζοντας τεχνικές όπως αυτές που προτείνονται στο [45].

Τέλος, δεδομένου ότι η μέθοδος της επίλυσης χαρακτηρίζεται από τη δυνατότητα γενίκευσης, μία ιδιαίτερα ενδιαφέρουσα εξέλιξη της συγκεκριμένης εργασίας είναι ο σχεδιασμός ενός αλγορίθμου που αφορά γλώσσες που δεν περιορίζονται σε Horn-λογικές, όπως για παράδειγμα η $\mathcal{ALC}$ που μπορεί να επαναγραφεί σε *διαζευκτική* (*disjunctive*) Datalog [95].

Παράρτημα Α΄

Ερωτήματα Αξιολόγησης

Ερωτήματα στην DL-Lite

OBO Protein

$$
\begin{array}{lcl}
Q(x) & \leftarrow & \text{has\_part}(x,y) \wedge \text{CHEBI\_23367}(y) \\
Q(x) & \leftarrow & \text{lacks\_part}(x,y) \wedge \text{SO\_0000418}(y) \\
Q(x) & \leftarrow & \text{PR\_000000001}(x) \\
Q(x) & \leftarrow & \text{derives\_from}(x,y) \wedge \text{PR\_000000001}(y) \\
Q(x) & \leftarrow & \text{has\_part}(x,y) \wedge \text{CHEBI\_23367}(y) \wedge \text{lacks\_part}(x,z) \wedge \text{SO\_0000418}(z)
\end{array}
$$

NCI

$$
\begin{array}{lcl}
Q(x) & \leftarrow & \text{Techniques}(x) \wedge \text{rTechnique\_Has\_Purpose}(x,y) \wedge \text{Detection}(y) \\
Q(x) & \leftarrow & \text{Gene}(x) \\
Q(x) & \leftarrow & \text{Gene}(x) \wedge \text{rGene\_Associated\_With\_Disease}(x,y) \wedge \text{Carcinoma}(y) \\
Q(x) & \leftarrow & \text{Gene}(x) \wedge \text{rGene\_Plays\_Role\_}in\text{\_Process}(x,y) \wedge \text{Pathogenesis}(y) \\
Q(x) & \leftarrow & \text{Immunoprotein\_Gene}(x) \wedge \text{rGene\_Plays\_Role\_in\_Process}(x,y) \wedge \\
 & & \text{Immune\_Response}(y) \wedge \text{rGene\_is\_Biomarker\_of}(x,z) \wedge \\
 & & \text{Breast\_Carcinoma}(z)
\end{array}
$$

OpenGALEN2

$$
\begin{array}{lcl}
Q(x) & \leftarrow & \text{Heme}(x) \\
Q(x) & \leftarrow & \text{BodyProcess}(x) \\
Q(x) & \leftarrow & \text{isConsequenceOf}(x,y) \wedge \text{Hypertension}(y) \\
Q(x) & \leftarrow & \text{Protein}(x) \\
Q(x) & \leftarrow & \text{isSpecificConsequenceOf}(x,y) \wedge \text{InfectionProcess}(y)
\end{array}
$$

P5X

$$
\begin{array}{lcl}
Q(x) & \leftarrow & \text{edge}(x,y) \\
Q(x) & \leftarrow & \text{edge}(x,y) \wedge \text{edge}(y,z) \\
Q(x) & \leftarrow & \text{edge}(x,y) \wedge \text{edge}(y,z) \wedge \text{edge}(z,k) \\
Q(x) & \leftarrow & \text{edge}(x,y) \wedge \text{edge}(y,z) \wedge \text{edge}(z,k) \wedge \text{edge}(k,l) \\
Q(x) & \leftarrow & \text{edge}(x,y) \wedge \text{edge}(y,z) \wedge \text{edge}(z,k) \wedge \text{edge}(k,l) \wedge \text{edge}(l,v)
\end{array}
$$

UX

$$
\begin{aligned}
Q(x) &\leftarrow \text{worksFor}(x,y) \wedge \text{affiliatedOrganizationOf}(y,z)\\
Q(x) &\leftarrow \text{Person}(x) \wedge \text{teacherOf}(x,y) \wedge \text{Course}(y)\\
Q(x) &\leftarrow \text{Student}(x) \wedge \text{advisor}(x,y) \wedge \text{FacultyStaff}(y) \wedge\\
&\quad\ \text{takesCourse}(x,z) \wedge \text{teacherOf}(y,z) \wedge \text{Course}(z)\\
Q(x) &\leftarrow \text{Person}(x) \wedge \text{worksFor}(x,y) \wedge \text{Organization}(y)\\
Q(x) &\leftarrow \text{Person}(x) \wedge \text{worksFor}(x,y) \wedge \text{University}(y) \wedge \text{hasAlumnus}(y,x)
\end{aligned}
$$

AX

$$
\begin{aligned}
Q(x) &\leftarrow \text{Device}(x) \wedge \text{assistsWith}(x,y)\\
Q(x) &\leftarrow \text{Device}(x) \wedge \text{assistsWith}(x,y) \wedge \text{UpperLimbMobility}(y)\\
Q(x) &\leftarrow \text{Device}(x) \wedge \text{assistsWith}(x,y) \wedge\\
&\quad\ \text{Hear}(y) \wedge \text{affects}(z,y) \wedge \text{Autism}(z)\\
Q(x) &\leftarrow \text{Device}(x) \wedge \text{assistsWith}(x,y) \wedge \text{PhysicalAbility}(y)\\
Q(x) &\leftarrow \text{Device}(x) \wedge \text{assistsWith}(x,y) \wedge \text{PhysicalAbility}(y) \wedge \text{affects}(z,y) \wedge\\
&\quad\ \text{Quadriplegia}(z)
\end{aligned}
$$

Ερωτήματα στην $\mathcal{ELHI}$

OBO Protein

$$
\begin{aligned}
Q(x) &\leftarrow \text{has\_part(x, y)} \wedge \text{CHEBI\_23367}(y)\\
Q(x) &\leftarrow \text{lacks\_part}(x,y) \wedge \text{SO\_0000418}(y)\\
Q(x) &\leftarrow \text{PR\_000000001}(x)\\
Q(x) &\leftarrow \text{derives\_from}(x,y) \wedge \text{PR\_000000001}(y)\\
Q(x) &\leftarrow \text{has\_part}(x,y) \wedge \text{CHEBI\_23367}(y) \wedge\\
&\quad\ \text{lacks\_part}(x,z) \wedge \text{SO\_0000418}(z)
\end{aligned}
$$

NASA SWEET

$$
\begin{aligned}
Q(x) &\leftarrow \text{Ocean}(x) \wedge \text{contains}(x,y) \wedge \text{Volcano}(y)\\
Q(x,y) &\leftarrow \text{City(x)} \wedge \text{inside(x, y)} \wedge \text{Country(y)} \wedge \text{isAdjacentTo}(x,z) \wedge \text{Desert}(z)\\
Q(x) &\leftarrow \text{Country}(x) \wedge \text{isPartOf}(x,y) \wedge \text{Continent}(y) \wedge \text{surroundedBy\_2D}(y,z) \wedge\\
&\quad\ \text{Ocean}(z)\\
Q(x) &\leftarrow \text{Region}(x)\\
Q(x,y) &\leftarrow \text{Island}(x) \wedge \text{isPart}(x,y) \wedge \text{Region}(y) \wedge \text{contains}(y,z) \wedge \text{Volcano}(z)
\end{aligned}
$$

PERIODIC Table

$$
\begin{aligned}
Q(x) &\leftarrow \text{Molecule}(x)\\
Q(x) &\leftarrow \text{Ion}(x)\\
Q(x) &\leftarrow \text{Atom}(x)\\
Q(x) &\leftarrow \text{MoleOfChemical}(x) \wedge \text{hasColor}(x,z) \wedge \text{WhiteColor}(z)\\
Q(x) &\leftarrow \text{MoleOfChemical}(x) \wedge \text{hasState}(x,y) \wedge \text{SolidState}(y)
\end{aligned}
$$

GALEN-Doctored,NotGALEN

$$
\begin{array}{lcl}
Q(x) & \leftarrow & \text{Heme}(x) \\
Q(x) & \leftarrow & \text{BodyProcess}(x) \\
Q(x) & \leftarrow & \text{IsCausallyLinkedTo}(x, y) \wedge \text{Hypertension}(y) \\
Q(x) & \leftarrow & \text{Protein}(x) \\
Q(x) & \leftarrow & \text{hasCause}(x, y) \wedge \text{Infection}(y)
\end{array}
$$

OpenGALEN2

$$
\begin{array}{lcl}
Q(x) & \leftarrow & \text{Heme}(x) \\
Q(x) & \leftarrow & \text{BodyProcess}(x) \\
Q(x) & \leftarrow & \text{isConsequenceOf}(x, y) \wedge \text{Hypertension}(y) \\
Q(x) & \leftarrow & \text{Protein}(x) \\
Q(x) & \leftarrow & \text{isSpecificConsequenceOf}(x, y) \wedge \text{InfectionProcess}(y)
\end{array}
$$

ExtendedDNS

$$
\begin{array}{lcl}
Q(x) & \leftarrow & \text{particular}(x) \\
Q(x) & \leftarrow & \text{participant}(x, y) \wedge \text{agent}(y) \\
Q(x) & \leftarrow & \text{proper-part}(x, y) \wedge \text{modal-description}(y) \\
Q(x) & \leftarrow & \text{d-uses}(x, y) \wedge \text{task}(y) \wedge \text{d-uses}(x, z) \wedge \text{description}(z) \\
Q(x) & \leftarrow & \text{part-of}(x, y) \wedge \text{region}(y)
\end{array}
$$

Ερωτήματα στην Horn$\mathcal{SHIQ}$

NASA SWEET

$$
\begin{array}{rcl}
Q(x) & \leftarrow & \text{Ocean}(x) \wedge \text{contains}(x, y) \wedge \text{Volcano}(y) \\
Q(x, y) & \leftarrow & \text{City(x)} \wedge \text{inside}(x, y) \wedge \text{Country}(y) \wedge \text{isAdjacentTo}(x, z) \wedge \text{Desert}(z) \\
Q(x) & \leftarrow & \text{Country}(x) \wedge \text{isPartOf}(x, y) \wedge \text{Continent}(y) \wedge \text{surroundedBy\_2D}(y, z) \wedge \\
 & & \text{Ocean}(z) \\
Q(x) & \leftarrow & \text{Region}(x) \\
Q(x, y) & \leftarrow & \text{Island}(x) \wedge \text{isPart}(x, y) \wedge \text{Region}(y) \wedge \text{contains}(y, z) \wedge \text{Volcano}(z)
\end{array}
$$

Periodic_full

$$
\begin{array}{lcl}
Q(x) & \leftarrow & \text{Molecule}(x) \\
Q(x) & \leftarrow & \text{Ion}(x) \\
Q(x) & \leftarrow & \text{Atom}(x) \\
Q(x) & \leftarrow & \text{MoleOfCompoundChemical}(x) \wedge \text{hasColor(x,z)} \wedge \text{WhiteColor}(z) \\
Q(x) & \leftarrow & \text{MoleOfCompoundChemical}(x) \wedge \text{hasState}(x, y) \wedge \text{SolidState}(y)
\end{array}
$$

DOLCE2.1

$$
\begin{aligned}
Q(x) &\leftarrow \text{particular}(x)\\
Q(x) &\leftarrow \text{participant}(x,y) \wedge \text{agent}(y)\\
Q(x) &\leftarrow \text{proper-part}(x,y) \wedge \text{modal-description}(y)\\
Q(x) &\leftarrow \text{d-uses}(x,y) \wedge \text{task}(y) \wedge \text{d-uses}(x,z) \wedge \text{description}(z)\\
Q(x) &\leftarrow \text{part-of}(x,y) \wedge \text{region}(y)
\end{aligned}
$$

UOBM

$$
\begin{aligned}
Q(x) &\leftarrow \text{worksFor}(x,y) \wedge \text{isAffiliatedOrganizationOf}(y,z)\\
Q(x) &\leftarrow \text{Postdoc}(x) \wedge \text{worksFor}(x,y) \wedge \text{University}(y) \wedge \text{hasAlumnus}(y,x)\\
Q(x) &\leftarrow \text{Person}(x) \wedge \text{like}(x,y) \wedge \text{Chair}(z) \wedge\\
&\quad\ \text{isHeadOf}(z,v) \wedge \text{like}(z,y)\\
Q(x) &\leftarrow \text{takesCourse}(x,y) \wedge \text{Professor}(z) \wedge \text{GraduateCourse}(y) \wedge\\
&\quad\ \text{isTaughtBy}(y,z)\\
Q(x,y) &\leftarrow \text{LeisureStudent}(x) \wedge \text{takesCourse}(x,y) \wedge\\
&\quad\ \text{isStudentOf}(x,z) \wedge \text{CSCourse}(y) \wedge \text{University}(z)\\
Q(x,y) &\leftarrow \text{enrollIn}(x,y) \wedge \text{hasDegreeFrom}(x,y) \wedge \text{University}(y)\\
Q(x,y) &\leftarrow \text{PeopleWithManyHobbies}(x) \wedge \text{isMemberOf}(x,y) \wedge \text{like}(x,z) \wedge\\
&\quad\ \text{TennisClass}(z) \wedge \text{hasMember}(y,w) \wedge \text{like}(w,z)\\
Q(x,y) &\leftarrow \text{GraduateCourse}(x) \wedge \text{isTaughtBy}(x,y) \wedge \text{isHeadOf}(y,z)\\
Q(x,y,z) &\leftarrow \text{Student}(x) \wedge \text{hasDegreeFrom}(x,y) \wedge \text{Professor}(z) \wedge\\
&\quad\ \text{worksFor}(z,x) \wedge \text{isAdvisorOf}(z,x)\\
Q(x,w,y) &\leftarrow \text{Professor}(x) \wedge \text{hasPublication}(x,z) \wedge \text{worksFor}(x,y) \wedge\\
&\quad\ \text{Dean}(w) \wedge \text{isMemberOf}(w,y) \wedge \text{isPublicationOf}(z,w)
\end{aligned}
$$

Bio-PAX

$$
\begin{aligned}
Q(x) &\leftarrow \text{CELLTYPE}(y,x) \wedge \text{externalReferenceUtilityClass}(y)\\
Q(x) &\leftarrow \text{CELLTYPE}(x,y) \wedge \text{openControlledVocabulary}(y)\\
Q(x) &\leftarrow \text{CELLTYPE}(y,x) \wedge \text{bioSource}(y)\\
Q(x) &\leftarrow \text{TISSUE}(y,x) \wedge \text{externalReferenceUtilityClass}(y)\\
Q(x) &\leftarrow \text{CONTROLLED}(y,x) \wedge \text{control}(y)\\
Q(x) &\leftarrow \text{CONTROLLER}(y,x) \wedge \text{control}(y)\\
Q(x) &\leftarrow \text{CELLULAR-LOCATION}(y,x) \wedge \text{utilityClass}(y)\\
Q(x) &\leftarrow \text{CELLULAR-LOCATION}(x,y) \wedge \text{openControlledVocabulary}(y)\\
Q(x) &\leftarrow \text{CELLULAR-LOCATION}(y,x) \wedge \text{physicalEntityParticipant}(y)\\
Q(x) &\leftarrow \text{CONFIDENCE}(x,y) \wedge \text{confidence}(y)
\end{aligned}
$$

Παράρτημα Β′

Γλωσσάριο Συμβόλων

$\mathbf{C}$	:	Το σύνολο των ατομικών εννοιών μιας ΠΛ γλώσσας
$\mathbf{R}$	:	Το σύνολο των ατομικών ρόλων μιας ΠΛ γλώσσας
$\mathbf{I}$	:	Το σύνολο των ατόμων μιας ΠΛ γλώσσας
$\sqcap$	:	Το σύμβολο της τομής δύο εννοιών
$\sqcup$	:	Το σύμβολο της ένωσης δύο εννοιών
$\exists$	:	Ο υπαρξιακός προσοδείκτης που χρησιμοποιείται για τον υπαρξιακό περιορισμό στις ΠΛ
$\forall$	:	Ο καθολικός προσοδείκτης τπου χρησιμοποιείται για τον καθολικό περιορισμό στις ΠΛ
$\leq$	:	Ο ελάχιστος περιορισμός πληθυκότητας στις ΠΛ
$\sqsubseteq$	:	Το σύμβολο της υπαγωγής δυο εννοιών ή ρόλων
$\subseteq$	:	Το σύμβολο του υποσυνόλου
$\neg$	:	Το σύμβολο της άρνησης μιας έννοιας, ή γενικότερα ενός στοιχείου
$\bot$	:	Η κενή έννοια
$\top$	:	Η καθολική έννοια
$\approx$	:	Το κατηγόρημα ισότητας
$\bowtie$	:	Αντιπροσωπεύει την ισότητα $\approx$, ή την ανισότητα
$\models$	:	Το σύμβολο της λογικής συνεπαγωγής (entailment)
$[]$	:	Συμβολίζει την κενή πρόταση
P^{-}	:	Ο αντίστροφος του ρόλου P
$\mathsf{Tra}(P)$	:	Αξιώμα που δηλώνει ότι ο ρόλος P είναι μεταβατικός
$\mathcal{T}$	:	Ένα σώμα ορολογίας
$\mathcal{A}$	:	Ένα σώμα ισχυρισμών
$\mathcal{Q}$	:	Ένα συζευκτικό ερώτημα
Δ^{I}	:	Το πεδίο διερμηνείας μίας ΠΛ
$\mathsf{body}(\mathcal{C})$	:	Το σώμα μίας πρότασης $\mathcal{C}$
$\mathsf{bound}(\mathcal{C})$	:	Το σύνολο των δεσμευμένων μεταβλητών της πρότασης $\mathcal{C}$
$\mathsf{unbound}(\mathcal{C})$	:	Το σύνολο των μη-δεσμευμένων μεταβλητών της πρότασης $\mathcal{C}$

$\mathsf{var}(\mathcal{C})$	:	Το σύνολο των μεταβλητών της πρότασης $\mathcal{C}$
$\mathsf{avar}(\mathcal{Q})$	:	Μία ορισμένη απάντηση του ερωτήματος $\mathcal{Q}$
$\mathsf{cert}(\mathcal{Q}, \mathcal{T}, \mathcal{A})$	:	Το σύνολο των απαντήσεων ενός συζευκτικού ερωτήματος $\mathcal{Q}$ με βάση τα $\mathcal{T}, \mathcal{A}$
$\mathcal{R}$	:	Σύνολο προτάσεων επαναγραφής
$\mathcal{I}$	:	Το σύμβολο ενός λογισμού
$\vdash^{\mathcal{I}}$	:	Το σύμβολο για την παραγωγή μίας πρότασης με βάση ένα σύστημα κανόνων επίλυσης, ή αλλιώς ένα λογισμό, $\mathcal{I}$
$\mathcal{I}_{lite}$	:	Λογισμός επαναγραφής για την DL-Lite που υλοποιείται στο σύστημα Rapid
$\mathcal{I}_{\mathcal{EL}}$	:	Λογισμός επαναγραφής για την $\mathcal{ELHI}$ που υλοποιείται στο σύστημα Rapid
$\mathcal{I}_{\mathsf{REQ}}$	:	Λογισμός επαναγραφής για την $\mathcal{ELHI}$ που υλοποιείται στο σύστημα Requiem
$\mathcal{I}_{\mathrm{H}\mathcal{S}}$	:	Λογισμός επαναγραφής για την Horn-$\mathcal{SHIQ}$
$\mathcal{I}_{\mathrm{RH}\mathcal{S}}$	:	Λογισμός επαναγραφής για την Horn-$\mathcal{SHIQ}$ που υλοποιείται στο σύστημα Rapid
$\mathcal{I}_{\mathsf{SLD}}$	:	Μία SLD παραγωγή
$\mathcal{I}_{\mathsf{ESLD}}$	:	Μία επεκταμένη SLD παραγωγή
$\mathsf{Rapid\text{-}Lite}(\mathcal{Q}, \mathcal{T})$	:	Η επαναγραφή που παράγεται από τον $\mathcal{I}_{lite}$ για ένα TBox, $\mathcal{T}$ και ένα ΣΕ, $\mathcal{Q}$
$\mathsf{Rapid\text{-}EL}(\mathcal{Q}, \mathcal{T})$	:	Η επαναγραφή που παράγεται από τον $\mathcal{I}_{\mathcal{EL}}$ για ένα TBox, $\mathcal{T}$ και ένα ΣΕ, $\mathcal{Q}$
$\mathsf{Rapid\text{-}Hshiq}(\mathcal{Q}, \mathcal{T})$	:	Η επαναγραφή που παράγεται από τον $\mathcal{I}_{\mathrm{RH}\mathcal{S}}$ για ένα TBox, $\mathcal{T}$ και ένα ΣΕ, $\mathcal{Q}$
$\mathsf{RQR}(\mathcal{Q}, \mathcal{T})$	:	Η επαναγραφή που παράγεται από τον $\mathcal{I}_{\mathsf{REQ}}$ για ένα TBox, $\mathcal{T}$ και ένα ΣΕ, $\mathcal{Q}$

□

Βιβλιογραφία

[1] Serge Abiteboul, Richard Hull, and Victor Vianu. *Foundations of Databases*. Addison-Wesley, 1995.

[2] Andrea Acciarri, Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, Mattia Palmieri, and Riccardo Rosati. Quonto: querying ontologies. In *AAAI*, volume 5, pages 1670–1671, 2005.

[3] Diego Calvanese De Giacomo Giuseppe Antonella Poggi, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Linking data to ontologies. *Journal on Data Semantics*, X:133–173, 2008.

[4] Alessandro Artale, Diego Calvanese, Roman Kontchakov, and Michael Zakharyaschev. The dl-lite family and relations. *Journal of Artificial Intelligence Research (JAIR)*, 36:1–69, 2009.

[5] Franz Baader. Terminological cycles in a description logic with existential restrictions. In *Proceedings of the International Joint Conference on Artificial Inteligence (IJCAI)*, volume 3, pages 325–330, 2003.

[6] Franz Baader, Sebastian Brandt, and Carsten Lutz. Pushing the $\mathcal{EL}$ envelope. In *Proceedings of the 19th International Joint Conference on AI (IJCAI-05)*, 2005.

[7] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2003.

[8] Franz Baader, Bernhard Hollunder, Bernhard Nebel, Hans-Jürgen Profitlich, and Enrico Franconi. An empirical analysis of optimization techniques for terminological representation systems: or:'making kris get a move on'. 2011.

[9] Franz Baader, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider. *The Description Logic Handbook: Theory, implementation and applications*. Cambridge University Press, 2002.

[10] Franz Baader and Wayne Snyder. Unification theory. *Handbook of automated reasoning*, 1:445–532, 2001.

[11] Leo Bachmain and Harald Ganzinger. Equational reasoning in saturation based theorem proving. *Automated deduction a basis for applications*, 1:353–397, 1998.

[12] Leo Bachmair and Harald Ganzinger. Strict basic superposition. In *Automated Deduction—CADE-15*, pages 160–174. Springer, 1998.

[13] Leo Bachmair and Harald Ganzinger. Resolution theorem proving. In *Handbook of Automated Reasoning*, pages 19–99. Elsevier and MIT Press, 2001.

[14] Leo Bachmair, Harald Ganzinger, Christopher Lynch, and Wayne Snyder. Basic paramodulation. *Information and computation*, 121(2):172–192, 1995.

[15] J. Barwise. *Handbook of Mathematical Logic*. Studies in Logic and the Foundations of Mathematics. Elsevier Science, 1982.

[16] Sean Bechhofer, Frank van Harmelen, James Hendler, Ian Horrocks, Deborah L. McGuinness, Peter F. Patel-Schneider, and Lynn Andrea Stein eds. OWL Web Ontology Language Reference. URL `http://www.w3.org/TR/owl-ref/`, Feb 2004.

[17] Tim Berners-Lee. The World Wide Web - Past, Present and Future. *Journal of Digital Information*, 1(1), 1997.

[18] Tim Berners-Lee, Robert Cailliau, Ari Luotonen, Henrik Frystyk Nielsen, and Arthur Secret. The World-Wide Web. *Communications ACM*, 37(8):76–82, 1994.

[19] Tim Berners-Lee and Mark Fischetti. *Weaving the web - the original design and ultimate destiny of the World Wide Web by its inventor*. HarperBusiness, 2000.

[20] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web. *Scientific American*, 2001.

[21] Meghyn Bienvenu. On the complexity of consistent query answering in the presence of simple ontologies. In *AAAI*, 2012.

[22] Meghyn Bienvenu, Camille Bourgaux, and François Goasdoué. Explaining query answers under inconsistency-tolerant semantics over description logic knowledge bases.

[23] Meghyn Bienvenu, Carsten Lutz, and Frank Wolter. First-order rewritability of atomic queries in horn description logics. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI 2013)*, 2013.

[24] Meghyn Bienvenu and Riccardo Rosati. New inconsistency-tolerant semantics for robust ontology-based data access. *Description Logics*, 1014:53–64, 2013.

[25] A. Borgida. On the relative expressiveness of description logics and predicate logics. *Artificial Intelligence*, 82:353–367, 1996.

[26] Andrea Calì, Georg Gottlob, and Thomas Lukasiewicz. A general datalog-based framework for tractable query answering over ontologies. In *Proceedings of the Twenty-Eigth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS 2009*, pages 77–86, 2009.

[27] Andrea Calì, Georg Gottlob, Thomas Lukasiewicz, Bruno Marnette, and Andreas Pieris. Datalog+/-: A family of logical knowledge representation and query languages for new applications. In *Logic In Computer Science*, pages 228–242, 2010.

[28] Andrea Calì, Georg Gottlob, and Andreas Pieris. Advanced processing for ontological queries. *Proceedings of the Very Large DataBase Endowment*, 3(1):554–565, 2010.

[29] Andrea Calì, Georg Gottlob, and Andreas Pieris. Advanced processing for ontological queries. *Proceedings of the VLDB Endowment*, 3(1-2):554–565, 2010.

[30] Andrea Calì, Georg Gottlob, and Andreas Pieris. Query answering under non-guarded rules in datalog+/-. In *Web Reasoning and Rule Systems*, pages 1–17. Springer, 2010.

[31] Andrea Calì, Georg Gottlob, and Andreas Pieris. Query rewriting under non-guarded rules. In *Proceedings of the 4th Alberto Mendelzon International Workshop on Foundations of Data Management, Buenos Aires, Argentina, May 17-20, 2010*, 2010.

[32] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Data complexity of query answering in description logics. In *Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning (KR 06)*, pages 260–270, 2006.

[33] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Tractable reasoning and efficient query answering in description logics: The dl-lite family. *Journal of Automated reasoning*, 39(3):385–429, 2007.

[34] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, Antonella Poggi, Mariano Rodriguez-Muro, Riccardo Rosati, Marco Ruzzi, and Domenico Fabio Savo. The mastro system for ontology-based data access. *Semantic Web Journal*, 2(1):43–53, 2011.

[35] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Data complexity of query answering in description logics. *Journal of Artificial Intelligence*, 195:335–360, 2013.

[36] Diego Calvanese, Evgeny Kharlamov, Werner Nutt, and Dmitriy Zheleznyakov. Evolution of dl-lite knowledge bases. In *Proceedings of International Semantic Web Conference (1) (ISWC)*, pages 112–128, 2010.

[37] Diego Calvanese, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Dl-lite: Tractable description logics for ontologies. In *In Proceedings of Association for the Advancement of Artificial Intelligence (AAAI)*, pages 602–607, 2005.

[38] Stefano Ceri, Georg Gottlob, and Letizia Tanca. *Logic programming and databases.* Springer-Verlag New York, Inc., New York, NY, USA, 1990.

[39] Chin-Liang Chang and Richard C. T. Lee. *Symbolic logic and mechanical theorem proving.* Computer science classics. Academic Press, 1973.

[40] Alexandros Chortaras, Despoina Trivela, and Giorgos Stamou. Optimized query rewriting in OWL 2 QL. In *Proceedings of the of 23rd International Conference on Automated Deduction (CADE-23)*, pages 192–206, 2011.

[41] Roberto De Virgilio, Giorgio Orsi, Letizia Tanca, and Riccardo Torlone. Semantic data markets: a flexible environment for knowledge management. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 1559–1564. ACM, 2011.

[42] Julian Dolby, Achille Fokoue, Aditya Kalyanpur, Li Ma, Edith Schonberg, Kavitha Srinivas, and Xingzhi Sun. *Scalable grounded conjunctive query evaluation over large and expressive knowledge bases.* Springer, 2008.

[43] Thomas Eiter, Georg Gottlob, Magdalena Ortiz, and Mantas Šimkus. Query answering in the description logic horn-\ mathcal {SHIQ}. In *Logics in Artificial Intelligence*, pages 166–179. Springer, 2008.

[44] Thomas Eiter, Carsten Lutz, Magdalena Ortiz, and Mantas Šimkus. Query answering in description logics: The knots approach. In *Logic, Language, Information and Computation*, pages 26–36. Springer, 2009.

[45] Thomas Eiter, Magdalena Ortiz, Mantas Simkus, Trung-Kien Tran, and Guohui Xiao. Query rewriting for Horn-$\mathcal{SHIQ}$ plus rules. In *Proc. of AAAI 2012*, 2012.

[46] FaCT++. `http://owl.man.ac.uk/factplusplus/`, 2003.

[47] Christian G Fermüller, Alexander Leitsch, Ullrich Hustadt, and Tanel Tammet. Resolution decision procedures. In *Handbook of Automated Reasoning*, pages 1791–1849. Elsevier Science Publishers BV, 2001.

[48] Melvin Fitting. *First-order logic and automated reasoning (2. ed.)*. Graduate texts in computer science. Springer, 1996.

[49] Matthew L. Ginsberg, editor. *Readings in nonmonotonic reasoning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1987.

[50] Haridimos Kondylakis Dimitris Plexousakis Giorgos Flouris, Dimitris Manakanatas and Grigoris Antoniou. Ontology change: classification and survey. *Knowledge Engineering Review*, 23(2):117–152, 2008.

[51] Birte Glimm, Ian Horrocks, Boris Motik, Rob Shearer, and Giorgos Stoilos. A novel approach to ontology classification. *Web Semantics: Science, Services and Agents on the World Wide Web*, 14:84–101, 2012.

[52] Birte Glimm, Ian Horrocks, Boris Motik, Giorgos Stoilos, and Zhe Wang. Hermit: an owl 2 reasoner. *Journal of Automated Reasoning*, 53(3):245–269, 2014.

[53] Birte Glimm, Carsten Lutz, Ian Horrocks, and Ulrike Sattler. Conjunctive query answering for the description logic $\mathcal{SHIQ}$. *Journal of Artificial Intelligence Research (JAIR)*, 31:157–204, 2008.

[54] Georg Gottlob, Giorgio Orsi, and Andreas Pieris. Ontological queries: Rewriting and optimization. In *Proceedings of the 27th International Conference on Ddata Engineering 2011*, 2011.

[55] Georg Gottlob, Giorgio Orsi, and Andreas Pieris. Ontological queries: Rewriting and optimization. In *Data Engineering (ICDE), 2011 IEEE 27th International Conference on*, pages 2–13. IEEE, 2011.

[56] Georg Gottlob, Giorgio Orsi, and Andreas Pieris. Ontological query answering via rewriting. In *Proceedings of the 15th International Conference on Advances in Databases and Information Systems*, pages 1–18. Springer-Verlag, 2011.

[57] Georg Gottlob and Thomas Schwentick. Rewriting ontological queries into small nonrecursive datalog programs. In *Proceedings of the 13th International Conference on Principles of Knowledge Representation and Reasoning (KR 2012)*, 2012.

[58] Benjamin N Grosof, Ian Horrocks, Raphael Volz, and Stefan Decker. Description logic programs: Combining logic programs with description logic. In *Proceedings of the 12th international conference on World Wide Web*, pages 48–57. ACM, 2003.

[59] Volker Haarslev and Ralf Möller. RACER System Description. In *International Joint Conference on Automated Reasoning-01*, volume 2083, 2001.

[60] Stijn Heymans, Li Ma, Darko Anicic, Zhilei Ma, Nathalie Steinmetz, Yue Pan, Jing Mei, Achille Fokoue, Aditya Kalyanpur, Aaron Kershenbaum, et al. Ontology reasoning with large data repositories. In *Ontology Management*, pages 89–128. Springer, 2008.

[61] Ian Horrocks. *Optimising Tableaux Decision Procedures for Description Logics*. PhD thesis, University of Manchester, 1997.

[62] Ian Horrocks, Oliver Kutz, and Ulrike Sattler. The even more irresistible sroiq. *KR*, 6:57–67, 2006.

[63] Ian Horrocks and Peter F Patel-Schneider. Reducing owl entailment to description logic satisfiability. In *The Semantic Web-ISWC 2003*, pages 17–29. Springer, 2003.

[64] Ian Horrocks and Ulrike Sattler. A description logic with transitive and inverse roles and role hierarchies. *Journal of Logic and Computation*, 9(3):385–410, 1999.

[65] Ian Horrocks and Ulrike Sattler. A tableaux decision procedure for $\mathcal{SHOIQ}$. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI 2005)*, pages 448–453, 2005.

[66] Ian Horrocks, Ulrike Sattler, and Stephan Tobies. Reasoning with individuals for the description logic $\mathcal{SHIQ}$. In David McAllester, editor, *Proceedings of the 17th International Conference on Automated Deduction (CADE 2000)*, volume 1831 of *Lecture Notes in Computer Science*, pages 482–496. Springer, 2000.

[67] Ian Horrocks and Sergio Tessaris. Querying the semantic web: a formal approach. In *The Semantic Web—ISWC 2002*, pages 177–191. Springer, 2002.

[68] Ullrich Hustadt, Boris Motik, and Ulrike Sattler. Reducing $\mathcal{SHIQ}^-$ Description Logic to Disjunctive Datalog Programs. In *Proc. of KR 2004*, pages 152–162, 2004.

[69] Ullrich Hustadt, Boris Motik, and Ulrike Sattler. Data complexity of reasoning in very expressive description logics. In *IJCAI*, volume 5, pages 466–471, 2005.

[70] Ullrich Hustadt, Boris Motik, and Ulrike Sattler. Reasoning in description logics by a reduction to disjunctive datalog. *J. of Autom. Reas.*, 39(3):351–384, 2007.

[71] Ullrich Hustadt, Boris Motik, and Ulrike Sattler. Deciding expressive description logics in the framework of resolution. *Information & Computation*, 206(5):579–601, 2008.

[72] Ullrich Hustadt and Renate A Schmidt. Issues of decidability for description logics in the framework of resolution. In *Proceedings Automated Deduction in Classical and Non-Classical Logics*, pages 191–205. Springer, 2000.

[73] Martha Imprialou, Giorgos Stoilos, and Bernardo Cuenca Grau. Benchmarking ontology-based query rewriting systems. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI 2012)*, 2012.

[74] Jorgen Jørgensen. *A treatise of formal logic.* A Treatise of Formal Logic. Russell & Russell, 1962.

[75] Yevgeny Kazakov. *Saturation-Based Decision Procedures for Extensions of the Guarded Fragment.* PhD thesis, Universität des Saarlandes, Saarbrücken, Germany, March 2006.

[76] Yevgeny Kazakov. Consequence-driven reasoning for Horn-SHIQ ontologies. In *Proceedings of IJCAI 2009*, volume 9, pages 2040–2045, 2009.

[77] Stanislav Kikot, Roman Kontchakov, Vladimir Podolskii, and Michael Zakharyaschev. Exponential lower bounds and separation for query rewriting. In *Automata, Languages, and Programming*, pages 263–274. Springer, 2012.

[78] Stanislav Kikot, Roman Kontchakov, and Michael Zakharyaschev. On (in)tractability of OBDA with OWL 2 QL. In *Proceedings of the 24th International Workshop on Description Logics (DL 2011)*, 2011.

[79] Atanas Kiryakov, Barry Bishoa, Damyan Ognyanoff, Ivan Peikov, Zdravko Tashev, and Ruslan Velkov. The features of bigowlim that enabled the bbc's world cup website. In *Proceedings Workshop on Semantic Data Management (SemData)*, 2010.

[80] George J Klir and Bo Yuan. *Fuzzy sets and fuzzy logic.* Prentice Hall New Jersey, 1995.

[81] Roman Kontchakov, Carsten Lutz, David Toman, Frank Wolter, and Michael Zakharyaschev. The combined approach to query answering in DL-Lite. In *Proceedings of the 20th International Conference Principles of Knowledge Representation and Reasoning (KR 2010)*, 2010.

[82] Adila Krisnadhi and Carsten Lutz. Data complexity in the EL family of description logics. In *Proceedings of the International Workshop on Description Logics*, 2007.

[83] Adila Krisnadhi and Carsten Lutz. Data complexity in the el family of dls. In *Proceedings of the 20th International Workshop on Description Logics (DL2007), volume 250 of CEUR-WS.* Citeseer, 2007.

[84] Markus Krötzsch and Sebastian Rudolph. Conjunctive queries for el with role composition. *Calvanese et al.[12]*, 2007.

[85] Markus Krötzsch, Sebastian Rudolph, and Pascal Hitzler. ELP: Tractable Rules for OWL 2. In Amit Sheth, Steffen Staab, Mike Dean, Massimo Paolucci, Diana Maynard, Timothy Finin, and Krishnaprasad Thirunarayan, editors, *Proceedings of the 7th International Semantic Web Conference (ISWC 2008)*, volume 5318 of *LNCS*, pages 649–664. Springer, Oktober 2008.

[86] Markus Krötzsch, Sebastian Rudolph, and Pascal Hitzler. Complexity boundaries for Horn description logics. In *Proceedings of Association for the Advancement of Artificial Intelligence*, pages 22–26. AAAI Press, 2007.

[87] Domenico Lembo, Maurizio Lenzerini, Riccardo Rosati, Marco Ruzzi, and Domenico Fabio Savo. Query rewriting for inconsistent dl-lite ontologies. In *Web Reasoning and Rule Systems*, pages 155–169. Springer, 2011.

[88] John W. Lloyd. *Foundations of Logic Programming.* Springer-Verlag New York, Inc., New York, NY, USA, 1984.

[89] Carsten Lutz. The complexity of conjunctive query answering in expressive description logics. In *Proceedings of the 4th International Joint Conference on Automated Reasoning (IJCAR)*, pages 179–193, 2008.

[90] Carsten Lutz. The complexity of conjunctive query answering in expressive description logics. In Alessandro Armando, Peter Baumgartner, and Gilles Dowek, editors, *Proceedings of the 4th International Joint Conference on Automated Reasoning (IJCAR2008)*, number 5195 in LNAI, pages 179–193. Springer, 2008.

[91] Carsten Lutz, David Toman, and Frank Wolter. Conjunctive query answering in the description logic EL using a relational database system. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI 2009)*, pages 2070–2075, 2009.

[92] Roman Kontchakov Mariano Rodriguez-Muro and Michael Zakharyaschev. Query rewriting and optimisation with database dependencies in ontop. In *Proc. of the 26th Int. Workshop on Description Logics (DL 2013)*, pages 917–929, 2013.

[93] E. Mendelson. *Introduction to Mathematical Logic.* Chapman & Hall/CRC, 1997.

[94] José Mora, Riccardo Rosati, and Óscar Corcho. kyrie2: Query rewriting under extensional constraints in *ELHIO*. In *Proc. of ISWC*, pages 568–583, 2014.

[95] Boris Motik. Description logics and disjunctive datalog—more than just a fleeting resemblance? In *Proceedings of the 4th Workshop on Methods for Modalities (M4M-4)*, volume 194, pages 246–265, 2005.

[96] Boris Motik. *Reasoning in Description Logics using Resolution and Deductive Databases*. PhD thesis, Universität Karlsruhe (TH), Karlsruhe, Germany, January 2006.

[97] Boris Motik, Bernardo Cuenca Grau, Ian Horrocks, Zhe Wu, Achille Fokoue, and Carsten Lutz. OWL 2 Web Ontology Language Profiles. W3C Recommendation, 27 October 2009.

[98] Boris Motik, Ian Horrocks, and Su Myeon Kim. Delta-reasoner: A semantic web reasoner for an intelligent mobile platform. In *Proceedings of WWW*, pages 63–72, 2012.

[99] Boris Motik, Ulrike Sattler, and Rudi Studer. Query answering for OWL-DL with rules. *Journal of Web Semantics*, 3(1):41–60, 2005.

[100] Boris Motik, Rob Shearer, and Ian Horrocks. A Hypertableau Calculus for SHIQ. In Diego Calvanese, Enriso Franconi, Volker Haarslev, Domenico Lembo, Boris Motik, Sergio Tessaris, and Anny-Yasmin Turhan, editors, *Proceedings of the 20th Int. Workshop on Description Logics (DL 2007)*, pages 419–426, Brixen/Bressanone, Italy, June 8–10 2007. Bozen/Bolzano University Press.

[101] Robert Nieuwenhuis and Albert Rubio. Theorem proving with ordering and equality constrained clauses. *Journal of Symbolic Computation*, 19(4):321–351, 1995.

[102] Giorgio Orsi and Andreas Pieris. Optimizing query answering under ontological constraints. *Journal of VLDB Endowment*, 4(11):1004–1015, 2011.

[103] Magdalena Ortiz, Diego Calvanese, and Thomas Eiter. Data complexity of query answering in expressive description logics via tableaux. *Journal of Automated Reasoning*, 41(1):61–98, 2008.

[104] Magdalena Ortiz, Sebastian Rudolph, and Mantas Simkus. Query answering in the horn fragments of the description logics shoiq and sroiq. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, volume 22, page 1039, 2011.

[105] Magdalena Ortiz, Sebastian Rudolph, and Mantas Šimkus. Query answering in the horn fragments of the description logics SHOIQ and SROIQ. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Two*, IJCAI'11, pages 1039–1044. AAAI Press, 2011.

[106] Christos H. Papadimitriou. *Computational complexity.* Academic Internet Publ., 2007.

[107] Peter F. Patel-Schneider, Patrick Hayes, and Ian Horrocks. OWL Web Ontology Language Semantics and Abstract Syntax. Technical report, W3C, Feb. 2004. W3C Recommendation, URL `http://www.w3.org/TR/2004/REC-owl-semantics-20040210/`.

[108] Héctor Pérez-Urbina, Ian Horrocks, and Boris Motik. Efficient query answering for OWL 2. In *Proceedings of the International Semantic Web Conference (ISWC2009)*, pages 489–504, 2009.

[109] Héctor Pérez-Urbina, Boris Motik, and Ian Horrocks. Rewriting Conjunctive Queries under Description Logic Constraints. In Andrea Cal'i, Georg Gottlob, Laks V.S. Lakshmanan, and Davide Martinenghi, editors, *Proceedigns of the International Workshop on Logic in Databases (LID 2008)*, Rome, Italy, May 19–20 2008.

[110] Héctor Pérez-Urbina, Boris Motik, and Ian Horrocks. Tractable query answering and rewriting under description logic constraints. *Journal of Applied Logic*, 8(2):186–209, 2010.

[111] Ian Horrocks Boris Motik und Laurent Pierre Pierre Chaussecourte, Birte Glimm. The energy management adviser at EDF. In *Proc. of ISWC 2013*, 2013.

[112] Amir Pnueli. The temporal logic of programs. *Foundations of Computer Science, IEEE Annual Symposium on*, 0:46–57, 1977.

[113] Mariano Rodriguez-Muro and Diego Calvanese. High performance query answering over DL-Lite ontologies. In *Proceedings of the 13th International Conference on Principles of Knowledge Representation and Reasoning (KR 2012)*, 2012.

[114] Riccardo Rosati. On conjunctive query answering in el. In *Proceedings of the 2007 International Workshop on Description Logics (DL2007)*, 2007.

[115] Riccardo Rosati. On conjunctive query answering in sc. In *20th International Workshop on Description Logics DL'07*. Citeseer, 2007.

[116] Riccardo Rosati and Alessandro Almatelli. Improving query answering over DL-Lite ontologies. In *Proceedings of KR 2010*, 2010.

[117] M.S. Schauß and G. Smolka. Attributive concept descriptions with complements. *Artificial Intelligence*, 48(1):1–26, 1991.

[118] Manfred Schmidt-Schauß and Gert Smolka. Attributive concept descriptions with complements. *Artificial intelligence*, 48(1):1–26, 1991.

[119] H. Scholz. *Geschichte der Logik*. Geschichte der Philosophie in Längsschnitten. Juncker u. Dünnhaupt, 1931.

[120] Nigel Shadbolt, Tim Berners-Lee, and Wendy Hall. The Semantic Web Revisited. *IEEE Intelligent Systems*, 21(3):96–101, 2006.

[121] Rob Shearer and Ian Horrocks. Exploiting partial information in taxonomy construction. *The Semantic Web-ISWC 2009*, pages 569–584, 2009.

[122] Frantisek Simancik, Yevgeny Kazakov, and Ian Horrocks. Consequence-based reasoning beyond horn ontologies. In *Proc. of IJCAI 2011*, pages 1093–1098, 2011.

[123] Evren Sirin and Bijan Parsia. Sparql-dl: Sparql query for owl-dl. In *In 3rd OWL Experiences and Directions Workshop (OWLED-2007)*, 2007.

[124] Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, and Yarden Katz. Pellet: A practical OWL-DL reasoner. *Journal of Web Semantics*, 5:51–53, 2007.

[125] Markus Stocker and Michael Smith. Owlgres: A scalable owl reasoner. In *OWLED*, volume 432, 2008.

[126] Giorgos Stoilos. Hydrowl: A hybrid query answering system for owl 2 dl ontologies. In *Web Reasoning and Rule Systems*, pages 230–238. Springer, 2014.

[127] Giorgos Stoilos. Ontology-based data access using rewriting, owl 2 rl systems and repairing. In *The Semantic Web: Trends and Challenges*, pages 317–332. Springer, 2014.

[128] Despoina Trivela, Giorgos Stoilos, Alexandros Chortaras, and Giorgos Stamou. Optimising resolution-based rewriting algorithms for dl ontologies. In *Proceedings of the 26th Workshop on Description Logics (DL 2013), Ulm, Germany*, 2013.

[129] Despoina Trivela, Giorgos Stoilos, Alexandros Chortaras, and Giorgos Stamou. Resolution-based query rewriting for horn-$\mathcal{SHIQ}$ ontologies. In *Proceedings of the 28th Workshop on Description Logics (DL 2015), Athens, Greece*, 2015.

[130] Ilianna Kollia und Birte Glimm. SPARQL query answering over owl ontologies. *Journal of Artificial Intelligence Research (JAIR)*, 48:253–303, 2013.

[131] Moshe Y. Vardi. On the complexity of bounded-variable queries (extended abstract). In *Proceedings of the fourteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*, PODS '95, pages 266–276, New York, NY, USA, 1995. ACM.

[132] Tassos Venetis, Giorgos Stoilos, and Giorgos Stamou. Query extensions and incremental query rewriting for OWL 2 QL ontologies. *Journal on Data Semantics*, 3:1–23, 2014.

[133] Tassos Venetis, Giorgos Stoilos, and Giorgos Stamou. Query rewriting under query refinements. *Knowledge-Based Systems*, 56:36–48, 2014.

[134] Michal Walicki. Mathematical logic - an introduction. `http://www.ii.uib.no/~michal/und/i227/book/book.pdf`, 2005. Department of Informatics, University of Bergen.

[135] Sebastian Wandelt and Ralf Moeller. Distributed island-based query answering for expressive ontologies. In *Proceedings of the 5th international conference on Advances in Grid and Pervasive Computing*, pages 461–470. Springer-Verlag, 2010.

Ευρετήριο Όρων