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

Αλγόριθμοι Επανεγγραφής Τροποποιημένων Ερωτημάτων για Βατές Περιγραφικές Λογικές

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

του

ΑΝΑΣΤΑΣΙΟΥ Γ. ΒΕΝΕΤΗ
Διπλωματούχου Ηλεκτρολόγου Μηχανικού &
Μηχανικού Υπολογιστών Ε.Μ.Π. (2005)

Αθήνα, Ιανουάριος 2014

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

Αλγόριθμοι Επανεγγραφής Τροποποιημένων Ερωτημάτων για Βατές Περιγραφικές Λογικές

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

του

ΑΝΑΣΤΑΣΙΟΥ Γ. ΒΕΝΕΤΗ

Διπλωματούχου Ηλεκτρολόγου Μηχανικού &
Μηχανικού Υπολογιστών Ε.Μ.Π. (2005)

Συμβουλευτική Επιτροπή: Γιώργος Στάμου
Στέφανος Κόλλιας
Ανδρέας-Γεώργιος Σταφυλοπάτης

Εγκρίθηκε από την επταμελή εξεταστική επιτροπή την 21η Ιανουαρίου 2014.

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

Αθήνα, Ιανουάριος 2014

...

ΑΝΑΣΤΑΣΙΟΣ Γ. ΒΕΝΕΤΗΣ
Διδάκτωρ Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

"Learn from yesterday, live for today, hope for tomorrow.
The important thing is to not stop questioning."
Albert Einstein

Ευχαριστίες

Ολοκληρώνοντας τη διδακτορική μου διατριβή θα ήθελα να εκφράσω τις ειλικρινείς μου ευχαριστίες σε όλα τα άτομα που με στήριξαν ψυχολογικά, ηθικά και πρακτικά για την εκπόνηση της.

Αρχικά θα ήθελα να ευχαριστήσω τον κ. Γιώργο Στάμου, Επίκουρο Καθηγητή Ε.Μ.Π. που ήταν και ο επιβλέπων της διδακτορικής μου διατριβής για την πολύτιμη καθοδήγηση του, τη δυνατότητα που μου έδωσε να ασχοληθώ με τον τομέα της έρευνας καθώς και για την προθυμότητα του να βοηθήσει στην επίλυση των προβλημάτων που παρουσιάζονταν. Επιπλέον θα ήθελα να ευχαριστήσω τον κ. Στέφανο Κόλλια, Καθηγητή Ε.Μ.Π. και διευθυντή του Εργαστηρίου Ψηφιακής Επεξεργασίας Εικόνας, Βίντεο και Πολυμέσων, ο οποίος παρείχε την αμέριστη υποστήριξη του οποτεδήποτε αυτό χρειαζόταν.

Ιδιαίτερες ευχαριστίες οφείλω στον Δρ. Γιώργο Στοΐλο για την αμέριστη συμπαράσταση και τη γόνιμη συνεργασία που είχαμε καθόλη τη διάρκεια της διδακτορικής μου διατριβής. Οι καίριες παρατηρήσεις του και οι κατευθυντήριες γραμμές που μου έδινε ήταν για μένα πολύτιμες και εποικοδομητικές.

Επιπλέον θα ήθελα να ευχαριστήσω τα μέλη του Εργαστηρίου Ψηφιακής Επεξεργασίας Εικόνας, Βίντεο και Πολυμέσων για τη δημιουργία ενός ευχάριστου περιβάλλοντος εργασίας. Τους Δρ. Γιώργο Καρυδάκη, Δρ. Νίκο Σίμου, Δρ. Θεόφιλο Μαΐλη, Δρ. Στέλιο Αστεριάδη, Δρ. Γιώργο Τόλια, αλλά και τους Γιώργο Γουδέλη, Γιάννη Καλαντίδη, Χρήστο Βαρυτιμίδη, την Ελένη Τσαλαπάτη, τη Δέσποινα Τριβέλα, την Σαρλίν Χονδρού και κυρίως τη Θέμιδα Ζερβού η οποία παρείχε απλόχειρα τη βοήθεια της για την εύρυθμη λειτουργία του εργαστηρίου.

Τέλος θα ήθελα να αφιερώσω την εργασία μου και να ευχαριστήσω βαθύτατα τη σύντροφο μου Ιωάννα Βέρροιου για την αμέριστη συμπαράσταση της και την υπομονή της καθώς και την οικογένεια μου που όλα αυτά τα χρόνια αποτελεί κύριο αρωγό των προσπαθειών μου.

Αναστάσιος Βενέτης
Αθήνα, Ιανουάριος 2014

ΠΕΡΙΛΗΨΗ

Το πρόβλημα συλλογιστικής της απάντησης συζευκτικών ερωτημάτων πάνω από βάσεις γνώσης Περιγραφικών Λογικών, μέσω της επανεγγραφής ερωτημάτων, έχει παρουσιάσει ιδιαίτερη άνθιση τα τελευταία χρόνια. Δεδομένου ενός συζευκτικού ερωτήματος και μιας βάσης γνώσης (TBox και ABox) μια διαδικασία επανεγγραφής ερωτημάτων παράγει ένα νέο ερώτημα που ενσωματώνει τους περιορισμούς της βάσης γνώσης που περιέχονται στο TBox, έτσι ώστε για οποιοδήποτε ABox (σύνολο δεδομένων) η αποτίμηση του αρχικού ερωτήματος πάνω στο TBox και το ABox να μπορεί να υπολογιστεί με την αποτίμηση μόνο του νέου ερωτήματος πάνω στο ABox. Επειδή η πολυπλοκότητα απάντησης ερωτημάτων σε εκφραστικές Περιγραφικές Λογικές είναι απαγορευτική έχουν αναπτυχθεί γλώσσες Περιγραφικών Λογικών που είναι βατές, όπως η οικογένεια γλωσσών DL-Lite, η $\mathcal{EL}$ και η οικογένεια της Datalog$^{\pm}$ για τις οποίες έχει παρουσιαστεί πληθώρα αλγορίθμων/συστημάτων επανεγγραφής ερωτημάτων. Όμως, όλοι οι αλγόριθμοι που γνωρίζουμε εκτελούνται κάθε φορά από την αρχή χωρίς να λαμβάνουν υπ' όψιν και να εκμεταλλεύονται προηγούμενες εκτελέσεις, ακόμα και εάν διαδοχικά ερωτήματα έχουν πολύ μικρές διαφορές, κάτι το οποίο είναι ιδιαίτερα συχνό στο διαδίκτυο.

Στην παρούσα διατριβή μελετάμε το πρόβλημα της επανεγγραφής ερωτημάτων τα οποία έχουν τροποποιηθεί με διάφορους τρόπους. Οι τρόποι αυτοί αφορούν στην προσθήκη ή αφαίρεση διακεκριμένων μεταβλητών ή ατόμων. Πιο συγκεκριμένα, μελετάμε το πρόβλημα υπολογισμού της επανεγγραφής ενός τροποποιημένου ερωτήματος εκμεταλλευόμενοι την επανεγγραφή που έχει υπολογιστεί για το αρχικό ερώτημα, αποφεύγοντας έτσι να την υπολογίσουμε εξαρχής. Στη συνέχεια παρουσιάζουμε βελτιστοποιήσεις οι οποίες αυξάνουν σημαντικά την απόδοση των αλγορίθμων μας. Ακολούθως, μειώνουμε την εκφραστικότητα των οντολογιών που μελετάμε στη γλώσσα DL-Lite$_R$ έτσι ώστε να βελτιστοποιήσουμε περαιτέρω το πρόβλημα του υπολογισμού της επανεγγραφής ενός ερωτήματος στο οποίο έχει προστεθεί ένα άτομο. Αυτό που παρουσιάζει ιδιαίτερο ενδιαφέρον είναι πως οι τεχνικές μας θέτουν τις βάσεις για έναν πρωτότυπο επαυξητικό αλγόριθμο επανεγγραφής σταθερών ερωτημάτων για οντολογίες DL-Lite$_R$. Πιο συγκεκριμένα, το ερώτημα μπορεί να «αναλυθεί» στα άτομα του και στη συνέχεια να επεξεργαστούμε κάθε άτομο επαυξητικά. Παρουσιάζουμε αναλυτικούς αλγόριθμους καθώς και ένα σύνολο από βελτιστοποιήσεις οι οποίες όπως φαίνεται και από την πειραματική μας αξιολόγηση βελτιώνουν την απόδοση του συστήματος μας και το καθιστούν ταχύτερο από όλα τα γνωστά συστήματα.

Τέλος, στη διατριβή αυτή μελετάμε το πρόβλημα απάντησης ερωτημάτων πάνω από ένα δίκτυο οντολογιών. Ειδικότερα, μελετάμε πως η Ασαφής Συνολοθεωρία και

η Ασαφής Λογική μπορούν να χρησιμοποιηθούν για την απόδοση σημασιολογίας και την επικύρωση των αντιστοιχίσεων ανάμεσα σε δύο οντολογίες ώστε να μπορέσουμε να επιλύσουμε ασυνέπειες που μπορεί να προκύψουν από την αντιστοίχιση τους. Αφού λοιπόν επιλύσουμε τις ασυνέπειες αυτές χρησιμοποιούμε τις δύο αυτές οντολογίες για να κατασκευάσουμε μια νέα (ασαφή) οντολογία η οποία είναι συνεπής και το ABox της οποίας περιέχει αναθέσεις που έχουν προκύψει από την ερμηνεία των ασαφών αντιστοιχίσεων και από τις δύο αρχικές οντολογιές κάνοντας έτσι εφικτή την απάντηση ερωτημάτων.

ABSTRACT

The reasoning problem of answering conjunctive queries over Description Logic knowledge bases via query rewriting has gained a lot of attention in the last few years. Given a conjunctive query and a knowledge base (TBox and ABox) a query rewriting procedure computes a new query that incorporates the constraints of the knowledge base that are described in its TBox, such that for any ABox (dataset) the evaluation of the initial query over the TBox and the ABox can be computed by evaluating only the new query over the ABox. Because the computational complexity in expressive Description Logics is prohibitive tractable Description Logics languages have been developed, such as the DL-Lite family, $\mathcal{EL}$ and the Datalog$^{\pm}$ family for which there have been presented many rewriting algorithms/systems. However, all these algorithms run from scratch without taking into consideration previous runs, even if successive queries are pretty similar, which is a very common scenario in the Web.

In this thesis we study the problem of query rewriting for queries that have been refined in various ways. These are the addition and removal of distinguished variables or atoms. More precisely, we study the problem of computing the rewriting of a query that has been refined taking into account the initial query without having to recompute it from scratch. In the following we present various optimisations that drastically increase the performance of our algorithms. Moreover, we limit the ontology language expressivity to DL-Lite$_R$ in order to further optimise the problem of query rewriting for queries that have been refined by the addition of an atom. Interestingly, our approach implies a novel incremental algorithm for computing the rewriting of a fixed query. More precisely, the query can be 'decomposed' into its atoms and then incrementally process each atom. We present detailed algorithms as well as several optimisations that as depicted by our experimental evaluation improve the performance of our system and show that it is faster than all the other systems.

Finally in this thesis we study the problem of query answering over an ontology network. Namely, given two ontologies and a set of mappings among them we study how Fuzzy Sets and Fuzzy Logic can be used to provide semantics for the mappings so that we can resolve inconsistencies that might arise from them. After resolving these inconsistencies a new (fuzzy) ontology that is consistent is created with an ABox that contains assertions, interpreted using the fuzzy mappings, from both the initial ontologies.

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

Κατάλογος Αλγορίθμων

Κατάλογος Συντμήσεων

ανν	:	αν και μόνο αν
μ.β.τ.	:	με βάση το
ΠΛ	:	Περιγραφικές Λογικές
ΣΕ	:	Συζευκτικό Ερώτημα
ΣΣΕ	:	Σύνολο Συζευκτικών Ερωτημάτων
τ.ω.	:	τέτοιο ώστε
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
DL-Lite$_{core}$	:	Η γλώσσα Περιγραφικών Λογικών DL-Lite$_{core}$
DL-Lite$_F$	:	Η γλώσσα Περιγραφικών Λογικών DL-Lite$_{core}$ με αξιώματα συναρτησιακών ρόλων $\mathsf{func}(R)$
DL-Lite$_R$	:	Η γλώσσα Περιγραφικών Λογικών DL-Lite$_{core}$ με αξιώματα υπαγωγής ρόλων $R \sqsubseteq E$
$\mathcal{EL}$	:	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
OWL 2 QL	:	OWL 2 Query Language
$\mathcal{S}$	:	Η γλώσσα Περιγραφικών Λογικών $\mathcal{ALC}$ με αξιώματα μεταβατικών ρόλων $(\mathcal{ALC}_{R^+}) : \mathsf{Trans}(R)$
$\mathcal{SI}$	:	Η γλώσσα Περιγραφικών Λογικών S με αντίστροφους ρόλους $(\mathcal{I}) : R^-$
$\mathcal{SHI}$	:	Η γλώσσα Περιγραφικών Λογικών $\mathcal{SI}$ με αξιώματα υπαγωγής ρόλων $(\mathcal{H}) : R \sqsubseteq S$
$\mathcal{SHIN}$	:	Η γλώσσα Περιγραφικών Λογικών $\mathcal{SHI}$ με περιορισμούς πληθυκότητας $(\mathcal{N}) :\leq nR, \geq nR$
$\mathcal{SHIQ}$	:	Η γλώσσα Περιγραφικών Λογικών $\mathcal{SHI}$ με προσοντούχους περιορισμούς πληθυκότητας $(\mathcal{Q}) :\leq nR.C, \geq nR.C$
$\mathcal{SHOIN}$	:	Η γλώσσα Περιγραφικών Λογικών $\mathcal{SHIN}$ με ονοματικές έννοιες $(\mathcal{O}) : \{o\}$
$\mathcal{SHOIQ}$	:	Η γλώσσα Περιγραφικών Λογικών $\mathcal{SHI}$ με ονοματικές έννοιες $(\mathcal{O}) : \{o\}$ και προσοντούχους περιορισμούς πληθυκότητας $(\mathcal{Q}) :\leq nR.C, \geq nR.C$

$\mathcal{SROIQ}$	:	Η γλώσσα Περιγραφικών Λογικών $\mathcal{SHOIQ}$ με πολύπλοκα αξιώματα υπαγωγής ρόλων ($\mathcal{R}$) : $R_1 \ldots R_n \sqsubseteq S$, αξιώματα ανακλαστικών ($\mathsf{Ref}(R)$), μη-ανακλαστικών ($\mathsf{Irr}(R)$), συμμετρικών ($\mathsf{Sym}(R)$), αντί-συμμετρικών ($\mathsf{ASym}(R)$), ξένων ($\mathsf{Dis}(R, S)$) ρόλων, απλής άρνησης ρόλων $(a, b) : \neg R$ και αυτοπαθών εννοιών $\exists R.\mathsf{Self}$
SW	:	Semantic Web
TBox	:	Terminological Box
WWW	:	World Wide Web
W3C	:	World Wide Web Consortium

Μέρος I

Θεμέλια

Κεφάλαιο 1

Εισαγωγή

Ένα από τα προβλήματα με τα οποία ασχολείται η Επιστήμη των Υπολογιστών και πιο συγκεκριμένα ο τομέας της *Τεχνητής Νοημοσύνης* (*Artificial Intelligence*) είναι αυτό της *αναπαράστασης γνώσης και της συλλογιστικής* (*knowledge representation and reasoning*) [24]. Το πρόβλημα αυτό αφορά στον τρόπο με τον οποίο η γνώση που απαντάται σε ένα επιστημονικό πεδίο ή εφαρμογή πρέπει να αποτυπώνεται, να οργανώνεται και να διαχειρίζεται, έτσι ώστε μέσω της διαδικασίας της συλλογιστικής να είναι εφικτή η διενέργεια σύνθετων εργασιών, που πλησιάζουν σε ποιότητα αυτές της ανθρώπινης σκέψης, και έχουν ως αποτέλεσμα την εξαγωγή χρήσιμων συμπερασμάτων. Το πρόβλημα αυτό έχει απασχολήσει από αρχαιοτάτων χρόνων πολλούς μαθηματικούς και φιλοσόφους πρώτος από τους οποίους ο Αριστοτέλης έθεσε τις βάσεις για αυτό που στις μέρες μας ονομάζεται *τυπική λογική* (*formal logic*) [81, 135].

Η τυπική λογική αφορά στην μελέτη μεθόδων εξαγωγής συμπερασμάτων με τη χρήση ενός τυπικού φορμαλισμού. Με την έννοια του τυπικού φορμαλισμού εννοούμε ένα καλώς ορισμένο *συντακτικό* (*syntax*) και μια καλώς ορισμένη *σημασιολογία* (*semantics*). Το *συντακτικό* αφορά στον ορισμό των συμβόλων που χρησιμοποιούνται από το φορμαλισμό καθώς και τους κανόνες με βάση τους οποίους τα σύμβολα αυτά είναι δυνατόν να συνδυαστούν. Από την άλλη, η *σημασιολογία* αφορά στον ορισμό των ερμηνειών που αποδίδονται στα σύμβολα αυτά αλλά και στους διάφορους συνδυασμούς τους που είναι επιτρεπτοί από το συντακτικό. Κατά τη διάρκεια των χρόνων έχει αναπτυχθεί μια πληθώρα από τυπικές λογικές και γλώσσες αναπαράστασης γνώσης όπως είναι η Λογική Πρώτης Τάξης (First Order Logic) [100, 167, 12], τα Σημασιολογικά Δίκτυα (Semantic Networks) [42], οι Τροπικές Λογικές (Modal Logics) [92], οι Μη-Μονότονες Λογικές (Non-Monotonic Logics) [55, 5], οι Χρονικές Λογικές (Temporal Logics) [119] και άλλες. Στην παρούσα διατριβή ασχολούμαστε με μια συγκεκριμένη οικογένεια φορμαλισμών που ονομάζονται *Περιγραφικές Λογικές* (*Description Logics*) [9] και χρησιμοποιούνται ως το λογικό υπόβαθρο για την γλώσσα ανάπτυξης οντολογιών OWL (Web Ontology Language) [14, 115], οι οποίες αποτελούν ακρογωνιαίο λίθο για τη έλευση του λεγόμενου *Σημασιολογικού Ιστού* (*Semantic Web–SW*) [19, 137]. Ο Σημασιολογικός Ιστός αποτελεί μια νέα μορφή του *Παγκόσμιου Ιστού* (*World Wide Web - WWW*) [17, 16, 18] στην οποία η πληροφορία οργανώνεται με τρόπο κατανοητό από τους υπολογιστές δίνοντας τη δυνατότητα σε διάφορους πράκτορες (agents) να φέρουν εις πέρας πολύπλοκες, ση-

μασιολογικά ορισμένες εργασίες, με ημι-αυτόματο τρόπο, παρέχοντας ταυτόχρονα σημασιολογική διαλειτουργικότητα.

Ο λόγος για τον οποίο οι Περιγραφικές Λογικές είναι ιδιαίτερα σημαντικές για το Σημασιολογικό Ιστό είναι ότι οι πρώτες αποτελούν ένα *εκφραστικό* και ταυτόχρονα *εύρωστα αποφασίσιμο* (*robustly decidable*) υποσύνολο της Λογικής Πρώτης Τάξης [22]. Αυτό έχει ως αποτέλεσμα τα συστήματα συλλογιστικής Περιγραφικών Λογικών [52, 146, 71, 105], που ονομάζονται αλλιώς και *μηχανές συλλογιστικής* (*reasoners*), να συμπεριφέρονται πολύ καλά σε πρακτικές εφαρμογές και να είναι σε θέση να διαχειριστούν μεγάλο όγκο πληροφορίας και γνώσης. Όπως είναι φυσιολογικό συνεπώς, χρησιμοποιούνται ευρέως σε διάφορες εφαρμογές, όπως σε πληροφοριακά συστήματα [72, 43], στη ρύθμιση παραμέτρων σε τηλεπικοινωνιακά συστήματα [15, 98], στην επεξεργασία πολυμεσικών κειμένων [13, 99, 142, 21, 118, 7, 113, 141, 143], στην ανάλυση εικόνων [102, 107] στη βιολογία [140, 150, 61, 136], στη γεωλογία [122, 95], στην ιατρική [62, 124], στη γεωγραφία [63, 96] και σε άλλες.

1.1 Περιγραφικές Λογικές

Οι Περιγραφικές Λογικές (ΠΛ) είναι μια οικογένεια φορμαλισμών που περιγράφουν τη γνώση ενός επιστημονικού πεδίου ή εφαρμογής με τη χρήση *ατόμων* (*individuals*), *εννοιών* (*concepts*) που αναπαριστούν σύνολα από άτομα, και *ρόλων* (*roles*) που αναπαριστούν δυαδικές σχέσεις ανάμεσα στα άτομα. Οι έννοιες και οι ρόλοι μπορεί να είναι *ατομικοί* (*atomic*) που ορίζονται ονομαστικά ή *σύνθετοι* (*complex*) που ορίζονται με τη χρήση κατασκευαστών πάνω σε έννοιες και ρόλους (ατομικούς ή σύνθετους). Οι κατασκευαστές αυτοί χαρακτηρίζουν την εκφραστικότητα κάθε ΠΛ. Για παράδειγμα χρησιμοποιώντας την ΠΛ $\mathcal{ALC}$ [134] μπορούμε να ορίσουμε την έννοια των μαθημάτων ενός πανεπιστημίου ως το σύνολο των προπτυχιακών και μεταπτυχιακών μαθημάτων, την έννοια των προπτυχιακών φοιτητών ως το σύνολο των φοιτητών που δεν παίρνουν κανένα μεταπτυχιακό μάθημα καθώς και ότι όποιος είναι συμφοιτητής με κάποιον τότε είναι φοιτητής, χρησιμοποιώντας τα παρακάτω αξιώματα:

$$\begin{aligned}
\textsf{Μάθημα} &\sqsubseteq \textsf{Προπτυχιακό} \sqcup \textsf{Μεταπτυχιακό} \\
\textsf{ΠροπτυχιακόςΦοιτητής} &\sqsubseteq \textsf{Φοιτητής} \sqcap \neg(\exists \textsf{παρακολουθεί}.\textsf{Μεταπτυχιακό}) \\
\exists \textsf{συμφοιτητήςΜε}.\top &\sqsubseteq \textsf{Φοιτητής}
\end{aligned}$$

όπου $\sqsubseteq$ ονομάζεται σχέση υπαγωγής, παρακολουθεί και συμφοιτητήςΜε είναι ρόλοι (δυαδικές σχέσεις) και $\exists$ ονομάζεται υπαρξιακός ποσοδείκτης (για την τυπική σημασιολογία των κατασκευαστών των εννοιών και των ρόλων ο αναγνώστης παραπέμπεται στην ενότητα 2.4).

Προσθέτοντας επιπλέον κατασκευαστές στην ΠΛ $\mathcal{ALC}$ μπορούμε να δημιουργήσουμε περισσότερο εκφραστικές γλώσσες. Εάν για παράδειγμα θέλουμε να εκφράσουμε ότι ο ρόλος συμφοιτητήςΜε είναι *μεταβατικός* (*transitive*) καθώς ότι οι ρόλοι διδάσκει και διδάσκεται είναι *αντίστροφοι* (*inverse*) τότε χρειαζόμαστε την ΠΛ $\mathcal{SI}$ [76] η οποία μπορεί να επεκταθεί περαιτέρω στην γλώσσα $\mathcal{SHI}$ [76] με χρήση της σχέσης

υπαγωγής ρόλων, επιτρέποντας δηλαδή αξιώματα υπαγωγής ρόλων, όπως το

παρακολουθείΠροπτυχιακό $\sqsubseteq$ παρακολουθεί

που δηλώνει ότι ο ρόλος παρακολουθείΠροπτυχιακό είναι *υπο-ρόλος* (*sub-role*) του ρόλου παρακολουθεί. Επιπλέον, επεκτείνοντας την $\mathcal{SHI}$ με *ονοματικές έννοιες* (*nominals*) ($\mathcal{O}$) όπως η έννοια {Πολυτεχνείο} και οι *περιορισμοί πληθυκότητας* (*number restrictions*) ($\mathcal{N}$) με τους οποίους μπορούμε να εκφράσουμε έννοιες λαμβάνοντας υπ' όψιν το πλήθος των συνδέσεων ενός ατόμου με άλλα σε μια σχέση, παίρνουμε την ΠΛ $\mathcal{SHOIN}$ [78]. Τέλος, ακόμα πιο εκφραστικές είναι οι γλώσσες $\mathcal{SHOIQ}$ [77] και $\mathcal{SROIQ}$ [75].

Το σύνολο των αξιωμάτων με βάση τα οποία ορίζονται νέες έννοιες και ρόλοι ορίζουν την *ορολογία* (*terminology*) ή αλλιώς *TBox* (*Terminological Box*) μιας ΠΛ και συμβολίζεται με $\mathcal{T}$. Εκτός όμως από τη δυνατότητα ορισμού σχέσεων μεταξύ εννοιών και ρόλων οι ΠΛ μας επιτρέπουν να κάνουμε και υποθέσεις όσον αφορά τον κόσμο που μοντελοποιούμε. Οι υποθέσεις αυτές γίνονται με τον καθορισμό *σχέσεων στιγμιοτύπου* (*instance relations*) ανάμεσα σε ένα άτομο (ζευγάρι ατόμων) και μια έννοια (ρόλο) και ονομάζονται *ισχυρισμοί* (*assertions*). Για παράδειγμα, εάν θεωρήσουμε τα άτομα Γιώργος, Μαθηματικά και Ιωάννα, μπορούμε να κάνουμε ισχυρισμούς πάνω στις έννοιες και τους ρόλους που εισάγαμε στο TBox της γνώσης γύρω από ένα πανεπιστήμιο, όπως Μάθημα(Μαθηματικά), Φοιτητής(Γιώργος), συμφοιτητήςΜε(Ιωάννα, Γιώργος), παρακολουθεί(Ιωάννα, Μαθηματικά) και παρακολουθεί(Γιώργος, Μαθηματικά). Το σύνολο των ισχυρισμών ορίζουν το *σώμα ισχυρισμών* (*assertional component*) ή αλλιώς *ABox* (*Assertion Box*) μιας ΠΛ και συμβολίζεται με $\mathcal{A}$. Τέλος, το σύνολο των αξιωμάτων που ορίζονται στο TBox και το σύνολο των ισχυρισμών που ορίζονται στο ABox αποτελούν μια *Βάση Γνώσης (ΒΓ)* (*Knowledge Base – KB*) ή αλλιώς μια *οντολογία* (*ontology*), που περιγράφει ένα συγκεκριμένο επιστημονικό πεδίο ή εφαρμογή.

1.2 Υπηρεσίες Συλλογιστικής

Πέρα από την μοντελοποίηση γνώσης για ένα επιστημονικό πεδίο ή μια εφαρμογή ένα σύστημα συλλογιστικής ΠΛ παρέχει υπηρεσίες συλλογιστικής όσον αφορά σε μια συγκεκριμένη οντολογία. Οι υπηρεσίες αυτές αφορούν στην εξαγωγή συμπερασμάτων (γνώσης) που δεν είναι ρητά δηλωμένα. Συνήθη προβλήματα συλλογιστικής που αφορούν στο TBox είναι ο καθορισμός εάν μια έννοια *υπάγει* (*subsumes*) μια άλλη–εάν δηλαδή το σύνολο των ατόμων που ανήκουν στην πρώτη ανήκουν κατ' ανάγκη και στη δεύτερη, ή εάν μια έννοια είναι *ικανοποιήσιμη* (*satisfiable*)–εάν δηλαδή υπάρχουν άτομα που να μπορούν να ανήκουν σε αυτή. Συνήθη προβλήματα συλλογιστικής που αφορούν στο ABox περιλαμβάνουν τον καθορισμό εάν ένα συγκεκριμένο άτομο αποτελεί στιγμιότυπο μιας έννοιας, ή εάν ένα ζεύγος από άτομα αποτελούν στιγμιότυπα ενός ρόλου. Τα προβλήματα συλλογιστικής αυτά τίθενται υπό μορφή ερώτησης στη μηχανή συλλογιστικής και μπορούν να αναχθούν σε μια ερώτηση ανάκτησης στιγμιοτύπων για μια έννοια ή ένα ρόλο. Παρόλα αυτά είναι ιδιαίτερα συχνό το φαινόμενο που οι χρήστες επιθυμούν να υποβάλλουν πιο σύνθετες ερωτήσεις οι οποίες να μη μπορούν να αναχθούν σε μια τέτοια ερώτηση ανάκτησης στιγμιοτύπων.

1.2.1 Απάντηση Ερωτημάτων

Η εκφραστικότητα για την υποβολή τέτοιων σύνθετων ερωτήσεων παρέχεται από τα *Συζευκτικά Ερωτήματα (ΣΕ)* (*Conjunctive Queries – CQ*) [1]. Για παράδειγμα, με βάση το TBox που έχουμε ορίσει προηγουμένως κάποιος θα μπορούσε να ζητήσει να ανακτηθούν δύο φοιτητές που παρακολουθούν ίδια μαθήματα, καθώς και τα μαθήματα αυτά. Το ερώτημα αυτό μπορεί να εκφραστεί με το παρακάτω Συζευκτικό Ερώτημα:

$$\textsf{Φοιτητής}(x) \wedge \textsf{Φοιτητής}(y) \wedge \textsf{παρακολουθεί}(x, z) \wedge \textsf{παρακολουθεί}(y, z) \rightarrow \mathcal{Q}_A(x, y, z)$$

όπου οι μεταβλητές x, y θα αντικατασταθούν με τα άτομα των φοιτητών και η μεταβλητή z με το άτομο του κοινού μαθήματος της οντολογίας, ενώ τα Φοιτητής και παρακολουθεί είναι μια έννοια και ένας ρόλος αντίστοιχα, που απαντώνται στην οντολογία. Η απάντηση του ερωτήματος αποτελείται από τις τριάδες ατόμων της οντολογίας για τις οποίες μετά την αντικατάσταση των μεταβλητών (x, y και z) με αυτές, οι ισχυρισμοί που προκύπτουν από τις έννοιες και τους ρόλους του ερωτήματος ισχύουν στην οντολογία. Έτσι χρησιμοποιώντας την οντολογία που έχουμε περιγράψει μέχρι στιγμής η τριάδα (Γιώργος, Ιωάννα, Μαθηματικά) είναι μια απάντηση. Πιο συγκεκριμένα, από την οντολογία δηλώνεται ρητά ότι ισχύουν οι ισχυρισμοί Φοιτητής(Γιώργος), παρακολουθεί(Γιώργος, Μαθηματικά) και παρακολουθεί(Ιωάννα, Μαθηματικά) ενώ με βάση το αξίωμα $\exists$συμφοιτητήςΜε.$\top \sqsubseteq$ Φοιτητής προκύπτει ο ισχυρισμός Φοιτητής(Ιωάννα), εφόσον ισχύει συμφοιτητήςΜε(Ιωάννα, Γιώργος).

Ο υπολογισμός των απαντήσεων ενός ερωτήματος με βάση μια οντολογία αποτελεί ένα ιδιαίτερα σημαντικό πρόβλημα συλλογιστικής που ονομάζεται *απάντηση ερωτημάτων* (*query answering*). Το πρόβλημα αυτό σχετίζεται άμεσα με την πραγματοποίηση εργασιών συλλογιστικής σε πολύ μεγάλα ABox και συνδέεται άρρηκτα με την περιοχή της *Πρόσβασης σε Δεδομένα πάνω από Οντολογίες* (*Ontology Based Data Access*) [120]. Η γενική ιδέα πίσω από την περιοχή αυτή είναι πως μια οντολογία χρησιμοποιείται για να παρέχει την εννοιολογική περιγραφή του πεδίου ενδιαφέροντος μιας εφαρμογής ενώ όλοι οι ισχυρισμοί της οντολογίας (δεδομένα) βρίσκονται αποθηκευμένοι σε μια *Βάση Δεδομένων (ΒΔ)* (*Database–DB*), διασφαλίζοντας έτσι πιο αποδοτική πρόσβαση σε αυτούς. Στο πλαίσιο αυτό η βάση δεδομένων θεωρείται πως δεν είναι πλήρης για κάποιο ερώτημα ως προς τη γνώση του κόσμου μας, εφόσον με χρήση της γνώσης αυτής μπορούν να προκύψουν νέοι ισχυρισμοί. Για το λόγο αυτό, και για να μπορέσουμε να τη «διορθώσουμε», χρησιμοποιούμε τα αξιώματα της εκάστοτε οντολογίας, που περιγράφουν τον κόσμο αυτό, ώστε να πραγματοποιηθούν διεργασίες συλλογιστικής πάνω στο ερώτημα και να παραχθούν όλες οι απαντήσεις που ικανοποιούν την οντολογία και τα δεδομένα.

Για το λόγο αυτό ο Motik [103] στην προσπάθεια του να μελετήσει αποδοτικούς αλγόριθμους συλλογιστικής σε μεγάλα ABox μελέτησε τη σχέση μεταξύ των ΠΛ και των επαγωγικών βάσεων δεδομένων (deductive databases) [37] και πρότεινε μια μέθοδο αναγωγής/μετατροπής μιας $\mathcal{SHIQ}$ οντολογίας σε ένα πρόγραμμα διαζευκτικής Datalog (disjunctive Datalog) η οποία βασίζεται στη μέθοδο της ανάλυσης (resolution) [10]. Επίσης έδειξε ότι τόσο η οντολογία όσο και το προκύπτων πρόγραμμα διαζευκτικής Datalog συνεπάγονται το ίδιο σύνολο από βασικά γεγονότα (ground

facts). Ο αλγόριθμος αυτός υλοποιήθηκε στο σύστημα KAON2[1] και έδειξε ότι βασικά προβλήματα συλλογιστικής των ΠΛ μπορούν να λυθούν με τη χρήση τεχνικών επαγωγικών βάσεων δεδομένων. Παρόμοιες εργασίες αποτελούν η δουλειά των Krötsch et al. [89], οι οποίοι πρότειναν ένα αλγόριθμο αναγωγής (reduction algorithm) για $\mathcal{ELP}$ οντολογίες σε κανόνες Datalog. Τέλος ο Kazakov [84] όρισε μια σειρά από μεθόδους ανάλυσης για ποικίλες Περιγραφικές Λογικές της οικογένειας $\mathcal{EL}$.

Οι παραπάνω εργασίες έδειξαν ότι τεχνικές που στηρίζονται στη μέθοδο της ανάλυσης (resolution) αλλά και στις (επαγωγικές) βάσεις δεδομένων μπορούν να χρησιμοποιηθούν αποτελεσματικά για την διεκπεραίωση υπηρεσιών συλλογιστικής στις Περιγραφικές Λογικές. Καθώς όμως η πολυπλοκότητα της συλλογιστικής σε πολύ εκφραστικές γλώσσες Περιγραφικών Λογικών είναι στην χειρότερη περίπτωση co-NP-πλήρης [93] *ως προς τα δεδομένα* (*data complexity*)–σε σχέση δηλαδή με το μέγεθος του ABox–η πολυπλοκότητα απάντησης Συζευκτικών Ερωτημάτων είναι απαγορευτική για πρακτικές εφαρμογές.

Για να είναι εφικτή η απάντηση Συζευκτικών Ερωτημάτων πάνω σε πολύ μεγάλα ABox είναι απαραίτητο οι αλγόριθμοι απάντησης να είναι *βατοί* (*tractable*) ως προς τα δεδομένα, όταν δηλαδή το ερώτημα και η οντολογία είναι δεδομένες. Για το λόγο αυτό αναπτύχθηκαν γλώσσες Περιγραφικών Λογικών που είναι βατές, όπως η οικογένεια γλωσσών DL-Lite [34, 33, 31], η $\mathcal{EL}$ [8] και η οικογένεια της Datalog$^{\pm}$ [25, 28]. Η ανάπτυξη των γλωσσών αυτών βασίστηκε στην ιδιότητα της *επανεγγραψιμότητας* (*rewritability*) για τις Περιγραφικές Λογικές. Η ιδιότητα αυτή ορίζει ότι δεδομένου ενός Συζευκτικού Ερωτήματος και μιας οντολογίας, το ερώτημα μπορεί να μετασχηματιστεί σε ένα νέο ερώτημα, που ονομάζεται *επανεγγραφή* (*rewriting*), το οποίο «περιέχει» τους περιορισμούς της οντολογίας. Πιο συγκεκριμένα, η οικογένεια γλωσσών της DL-Lite είναι *επανεγγράψιμη σε πρώτη-τάξη* (*first-order rewritability*) με το αρχικό Συζευκτικό Ερώτημα να μετασχηματίζεται σε μια επανεγγραφή Συνόλου Συζευκτικών Ερωτημάτων, ενώ οι οικογένειες της $\mathcal{EL}$ και της Datalog$^{\pm}$ είναι *επανεγγράψιμες σε Datalog* (*Datalog rewritable*) με το αρχικό Συζευκτικό Ερώτημα να μετασχηματίζεται στην χειρότερη περίπτωση σε ένα πρόγραμμα Datalog. Στη συνέχεια, για κάθε ABox, το οποίο όπως αναφέραμε και προηγουμένως μπορεί να είναι αποθηκευμένο και σε μια βάση δεδομένων, σχεσιακή ή επαγωγική αντίστοιχα, η απάντηση του ερωτήματος προκύπτει από την αποτίμηση του επανεγγραμμένου ερωτήματος πάνω στα δεδομένα. Για παράδειγμα επανεγγραφή του ερωτήματος

$$\textsf{Φοιτητής}(x) \wedge \textsf{Φοιτητής}(y), \textsf{παρακολουθεί}(x,z) \wedge \textsf{παρακολουθεί}(y,z) \rightarrow \mathcal{Q}_A(x,y,z)$$

που παρουσιάσαμε παραπάνω, με βάση το TBox $\mathcal{T} = \{\exists \textsf{συμφοιτητήςΜε}.\top \sqsubseteq \textsf{Φοιτητής}\}$ είναι το σύνολο των ερωτημάτων:

$$\begin{aligned}
\textsf{Φοιτητής}(x) \wedge \textsf{Φοιτητής}(y) \wedge \textsf{παρακολουθεί}(x,z) \wedge \textsf{παρακολουθεί}(y,z) &\rightarrow Q_A(x,y,z)\\
\textsf{Φοιτητής}(x) \wedge \textsf{συμφοιτητήςΜε}(y,z_1) \wedge \textsf{παρακολουθεί}(x,z) \wedge \textsf{παρακολουθεί}(y,z) &\rightarrow Q_A(x,y,z)\\
\textsf{συμφοιτητήςΜε}(x,z_2) \wedge \textsf{Φοιτητής}(y) \wedge \textsf{παρακολουθεί}(x,z) \wedge \textsf{παρακολουθεί}(y,z) &\rightarrow Q_A(x,y,z)\\
\textsf{συμφοιτητήςΜε}(x,z_2) \wedge \textsf{συμφοιτητήςΜε}(y,z_1) \wedge \textsf{παρακολουθεί}(x,z) \wedge \textsf{παρακολουθεί}(y,z) &\rightarrow Q_A(x,y,z)
\end{aligned}$$

με τις z_1, z_2 να είναι νέες μεταβλητές. Παρατηρούμε ότι το σύνολο των ερωτημάτων αυτών προκύπτει από την αντικατάσταση του ατόμου $\textsf{Φοιτητής}(k)$ με το άτομο

[1] `http://kaon2.semanticweb.org/`

$\textsf{συμφοιτητήςΜε}(k, z_i)$, όπου $k \in \{x, y\}$ και $i = \{1, 2\}$. Η διαδικασία για την κατασκευή της επανεγγραφής ενός ερωτήματος ονομάζεται *επανεγγραφή ερωτημάτων* (*query rewriting*) και αποτελεί το κύριο θέμα που θα μας απασχολήσει στη συνέχεια της διατριβής. Η πολυπλοκότητα για την αποτίμηση ερωτημάτων πρώτης-τάξης ανήκει στην κλάση πολυπλοκότητας AC_0 [112, 166, 31] που περιέχεται στην LOGSPACE, ενώ η πολυπλοκότητα για την αποτίμηση προγραμμάτων Datalog είναι PTIME-πλήρης [112, 166, 129, 88, 90], σε σχέση με τα δεδομένα. Συνεπώς, η απάντηση ερωτημάτων για λογικές που είναι επανεγγράψιμες είναι χαμηλής πολυπλοκότητας.

Τα τελευταία χρόνια η μελέτη της επανεγγραψιμότητας έχει παρουσιάσει ιδιαίτερη άνθιση τόσο από άποψη μελέτης πολυπλοκότητας [33, 67, 85, 109] όσο και στην κατασκευή διαφορετικών αλγορίθμων και συστημάτων επανεγγραφής ερωτημάτων [31, 116, 132, 40, 64, 108, 49, 127].

1.3 Παρουσίαση Προβλήματος

Όλοι οι αλγόριθμοι επανεγγραφής ερωτημάτων που γνωρίζουμε εκτελούνται κάθε φορά από την αρχή χωρίς να λαμβάνουν υπ' όψιν και να εκμεταλλεύονται προηγούμενες εκτελέσεις, ακόμα και αν διαδοχικά ερωτήματα έχουν πολύ μικρές διαφορές, το οποίο είναι πολύ συχνό φαινόμενο. Για παράδειγμα, σε διάφορα σενάρια στο Διαδίκτυο [19] έχει αποδειχθεί ότι οι χρήστες συνήθως θέτουν πρώτα ένα «γενικό» ερώτημα και στη συνέχεια, ανάλογα με την απάντηση που παίρνουν, το τροποποιούν προσθέτοντας ή αφαιρώντας κάποιους περιορισμούς, κάνοντας το ερώτημα τους πιο συγκεκριμένο ή πιο γενικό [80, 114, 79]. Συνεπώς, το τελικό ερώτημα ενδεχομένως να είναι γνωστό μόνο ύστερα από αρκετές τροποποιήσεις.

Για παράδειγμα, κάποιος χρήστης μπορεί αρχικά να ζητήσει από μια βάση δεδομένων με φοιτητές να ανακτηθούν όλοι οι φοιτητές χρησιμοποιώντας το παρακάτω συζευκτικό ερώτημα:

$$\textsf{Φοιτητής}(x) \rightarrow \mathcal{Q}_A(x)$$

Στη συνέχεια ο χρήστης μπορεί να τροποποιήσει την αναζήτηση του ζητώντας μόνο εκείνους τους φοιτητές που παρακολουθούν ένα οποιοδήποτε μάθημα, επεκτείνοντας το προηγούμενο ερώτημα με το άτομο $\textsf{παρακολουθεί}(x, y)$ θέτοντας έτσι το παρακάτω συζευκτικό ερώτημα:

$$\textsf{Φοιτητής}(x) \wedge \textsf{παρακουλουθεί}(x, y) \rightarrow \mathcal{Q}_A(x)$$

Ακολούθως, το νέο ερώτημα μπορεί να τροποποιηθεί επιπλέον ζητώντας εκτός από τους φοιτητές να ανακτηθεί και το μάθημα που παρακολουθούν. Αυτό μπορεί να γίνει με την προσθήκη της μεταβλητής y στις διακεκριμένες μεταβλητές (μεταβλητές απάντησης), οπότε έχουμε το παρακάτω συζευκτικό ερώτημα:

$$\textsf{Φοιτητής}(x) \wedge \textsf{παρακουλουθεί}(x, y) \rightarrow \mathcal{Q}_A(x, y)$$

Επιπλέον, ο χρήστης μπορεί να τροποποιήσει και πάλι το ερώτημα του ζητώντας να επιστραφούν μόνο τα μαθήματα τα οποία παρακολουθεί κάποιος φοιτητής, αφαιρώντας

δηλαδή την μεταβλητή x από τις διακεκριμένες μεταβλητές και θέτοντας το παρακάτω συζευκτικό ερώτημα:

$$\textsf{Φοιτητής}(x) \wedge \textsf{παρακουλουθεί}(x, y) \rightarrow \mathcal{Q}_A(y)$$

Τέλος, το νέο αυτό ερώτημα μπορεί να τροποιηθεί περαιτέρω ζητώντας να επιστραφούν τα μαθήματα που παρακολουθεί κάποιος χωρίς να είναι απαραίτητα φοιτητής, αφαιρώντας δηλαδή από το προηγούμενο ερώτημα το άτομο $\textsf{Φοιτητής}(x)$ και θέτοντας συνεπώς το παρακάτω συζευκτικό ερώτημα:

$$\textsf{παρακουλουθεί}(x, y) \rightarrow \mathcal{Q}_A(y)$$

Σε όλες αυτές τις περιπτώσεις όλα τα υπάρχοντα συστήματα επανεγγραφής παράγουν μια επανεγγραφή για κάθε ερώτημα που έχει τροποποιηθεί εκτελώντας κάθε φορά τον αλγόριθμο τους εξαρχής, χωρίς να χρησιμοποιήσουν την πληροφορία που μόλις έχουν παράγει, παρόλο που τα ερωτήματα αυτά παρουσιάζουν πολύ μικρές διαφορές μεταξύ τους.

Στην παρούσα διατριβή μελετάμε το πρόβλημα της επανεγγραφής ερωτημάτων που έχουν τροποποιηθεί με όλους τους τρόπους που περιγράψαμε παραπάνω χρησιμοποιώντας όσο το δυνατόν περισσότερο την επανεγγραφή που έχει κατασκευαστεί για το αρχικό ερώτημα. Πιο συγκεκριμένα, για ένα TBox $\mathcal{T}$, ένα ερώτημα $\mathcal{Q}$, και μια επανεγγραφή $\mathcal{R}$ που έχει υπολογιστεί προηγούμενα για τα $\mathcal{Q}, \mathcal{T}$ μελετάμε πώς μπορούμε να υπολογίσουμε μια επανεγγραφή για ένα νέο ερώτημα που προκύπτει από το $\mathcal{Q}$ προσθέτοντας ή αφαιρώντας κάποια μεταβλητή στις μεταβλητές απάντησης του ή κάποιο άτομο στο σώμα του, εκμεταλλευόμενοι όσο το δυνατόν περισσότερο τη δοθείσα επανεγγραφή $\mathcal{R}$, αποφεύγοντας να την υπολογίσουμε εξαρχής. Στη συνέχεια, για να βελτιώσουμε την απόδοση των αλγορίθμων αυτών παρουσιάζουμε μια σειρά από βελτιστοποιήσεις.

Αυτό που παρουσιάζει ιδιαίτερο ενδιαφέρον είναι το γεγονός πως οι τεχνικές μας για την επανεγγραφή ερωτημάτων που έχουν επεκταθεί με νέα άτομα θέτουν τις βάσεις για έναν πρωτότυπο (επαυξητικό) αλγόριθμο για την επανεγγραφή σταθερών ερωτημάτων. Πιο συγκεκριμένα, δεδομένου ενός (προκαθορισμένου) ερωτήματος μπορεί κανείς να επιλέξει ένα από τα άτομα του, να υπολογίσει μια επανεγγραφή για αυτό, και στη συνέχεια επαναληπτικά να προσθέσει τα υπόλοιπα άτομα επεκτείνοντας κάθε φορά την προηγουμένως υπολογισμένη επανεγγραφή. Όταν έχουμε επεξεργαστεί όλα τα άτομα τότε θα έχουμε υπολογίσει μια επανεγγραφή για το αρχικό ερώτημα. Βασιζόμενοι σε αυτή την ιδέα παρουσιάζουμε ένα αναλυτικό αλγόριθμο επανεγγραφής ερωτημάτων για συζευκτικά ερωτήματα πάνω από οντολογίες DL-Lite καθώς και ένα σύνολο από βελτιστοποιήσεις οι οποίες αυξάνουν σημαντικά την αποδοτικότητα του αλγορίθμου μας.

Αξίζει να σημειώσουμε ότι απ' όσο γνωρίζουμε αυτή είναι η πρώτη θεωρητική και πρακτική μελέτη του προβλήματος αυτού, τόσο στον χώρο των οντολογιών όσο και στον χώρο της θεωρίας βάσεων δεδομένων και πιο συγκεκριμένα στον χώρο της απάντησης ερωτημάτων με βάση περιορισμούς.

Τέλος, στην παρούσα διατριβή μελετάμε ένα άλλο πρόβλημα που συνδέεται άρρηκτα με την απάντηση ερωτημάτων. Πρόκειται για το πρόβλημα της απάντησης ερωτημάτων πάνω από ένα δίκτυο οντολογιών [121, 82, 57, 50, 139]. Είναι ιδιαίτερα

σύνηθες στο πλαίσιο του Σημασιολογικού Ιστού, να υπάρχουν οντολογίες που περιγράφουν το ίδιο, ή επικαλυπτόμενα πεδία ενδιαφέροντος, αλλά παρόλα αυτά να είναι ετερογενείς. Πιο συγκεκριμένα, μια οντότητα, έννοια ή ρόλος, μπορεί να ορίζεται με διαφορετικά ονόματα, αλλά και διαφορετικούς τρόπους σε δύο οντολογίες. Για παράδειγμα η έννοια του κινητού τηλεφώνου μπορεί να ορίζεται σε μια οντολογία από την έννοια ΚινητόΤηλέφωνο ενώ σε μια άλλη από την έννοια ΚινΤηλέφωνο, ή ακόμα και από την έννοια Τηλέφωνο $\sqcap$ ΦορητήΣυσκευή. Για να μπορέσει να επιτευχθεί *σημασιολογική διαλειτουργικότητα* (*semantic interoperability*) που είναι απαραίτητη στο πλαίσιο του Σημασιολογικού Ιστού, οι ετερογενείς αυτές οντολογίες πρέπει να αντιστοιχιστούν κατασκευάζοντας έτσι ένα δίκτυο οντολογιών. Παρόλα αυτά, οι περισσότερες τεχνικές αντιστοίχισης [50] δεν λαμβάνουν υπ' όψιν τη σημασιολογία των οντολογιών οδηγώντας έτσι στην κατασκευή αντιστοιχίσεων που προκαλούν ασυνέπειες. Με άλλα λόγια, οι αντιστοιχίσεις που προκύπτουν δεν μπορούν να ερμηνευτούν ως σημασιολογικές σχέσεις ανάμεσα στις οντότητες τους, γεγονός που είναι απαραίτητο για την *ολοκλήρωση οντολογιών* (*ontology integration*) και την απάντηση ερωτημάτων πάνω από ένα δίκτυο οντολογιών.

Για το λόγο αυτό τα τελευταία χρόνια έχουν υπάρξει διάφορες προσπάθειες για την επικύρωση των αντιστοιχίσεων χρησιμοποιώντας κυρίως πιθανοτικές προσεγγίσεις [30, 36]. Οι προσεγγίσεις αυτές όμως, υποφέρουν από περιορισμούς λόγω της φύσης των αντιστοιχίσεων και του τρόπου με τον οποίο υπολογίζονται οι πιθανότητες. Έτσι, στην διατριβή αυτή παρουσιάζουμε μια τελείως διαφορετική προσέγγιση για να μπορέσουμε να ερμηνεύσουμε τις αντιστοιχίσεις. Υποθέτοντας ότι μια αντιστοίχιση δηλώνει την ομοιότητα ανάμεσα σε δύο οντότητες, μπορούμε να αναθέσουμε τα άτομα που ανήκουν στην πρώτη έννοια, και στην δεύτερη με ένα συγκεκριμένο βαθμό, που είναι ακριβως η σημασιολογία των συναρτήσων συμμετοχής της Ασαφούς Συνολοθεωρίας [168, 86] και της Ασαφούς Λογικής [73]. Ο βαθμός συμμετοχής καθορίζεται από το βαθμό της σχέσης ομοιότητας της αντιστοίχισης, και υπολογίζεται από κάποια τεχνική αντιστοίχισης. Έτσι λοιπόν, χρησιμοποιώντας τις αντιστοιχίσεις που παράγονται για να κατασκευάσουμε ασαφείς ισχυρισμούς, παρέχουμε μια τυπική ερμηνεία των αντιστοιχίσεων. Επιπλέον, χρησιμοποιώντας το πλαίσιο των ασαφών ΠΛ [156, 157, 155] έχουμε τη δυνατότητα να εντοπίσουμε και να λύσουμε τυχόν ασυνέπειες.

1.4 Συνεισφορά και Δομή της Διατριβής

Η κύρια συνεισφορά της διατριβής μας αφορά στο πρόβλημα της επανεγγραφής ερωτημάτων. Τα αποτελέσματα μας σχετικά με το πρόβλημα αυτό έχουν πολλές θεωρητικές και πρακτικές συνέπειες και παρέχουν δυνατότητες για περαιτέρω μελλοντική έρευνα. Αρχικά δείχνουν ότι η επανεγγραφή ερωτημάτων τα οποία έχουν τροποποιηθεί μπορεί να πραγματοποιηθεί ιδιαίτερα αποδοτικά για όλες τις επανεγγράψιμες ΠΛ, χρησιμοποιώντας την πληροφορία, επανεγγραφή, που έχει ήδη υπολογιστεί για το αρχικό ερώτημα . Επιπλέον, όσον αφορά στην DL-Lite τα αποτελέσματα μας δείχνουν πως η επανεγγραφή σταθερών, προκαθορισμένων, ερωτημάτων μπορεί να πραγματο-

ποιηθεί σε μεγάλο βαθμό παράλληλα, κάτι το οποίο από όσο γνωρίζουμε δεν ήταν προτέρως γνωστό. Επιπρόσθετα, η ανεξάρτητη επεξεργασία των ατόμων έδειξε ότι ένα πολύ γνωστό πρόβλημα του πρωτότυπου αλγορίθμου για την DL-Lite [31], αυτό του βήματος μείωσης, μπορεί να βελτιωθεί σημαντικά. Ακολούθως, τα αποτελέσματα μας δίνουν τη δυνατότητα για την μελέτη αποδοτικών μεθόδων για την απάντηση ερωτημάτων, εφόσον ενδεχομένως να μπορούμε να εκμεταλλευτούμε τα πλεονεκτήματα που αποκομίζουμε από την επαυξητική επανεγγραφή ερωτημάτων. Τέλος, τα ιδιαίτερα ενθαρρυντικά αποτελέσματα για την αποδοτικότητα του επαυξητικού αλγόριθμου επανεγγραφής που παρουσιάζουμε δίνουν τη δυνατότητα για μελέτη αλγορίθμων που χρησιμοποιούν την επαυξητική αυτή τεχνική για επανεγγραφή ερωτημάτων για πιο εκφραστικές οντολογίες όπως $\mathcal{ELHI}$ και Horn-$\mathcal{SHIQ}$ σε επανεγγραφές Datalog και Disjunctive Datalog αντίστοιχα.

Τέλος, όσον αφορά στην απάντηση ερωτημάτων μέσω δικτύου οντολογιών η μελέτη μας δείχνει πως η Ασαφής Συνολοθεωρία και η Ασαφής Λογική μπορούν να χρησιμοποιηθούν για την απόδοση σημασιολογίας στις αντιστοιχίσεις ανάμεσα σε οντολογίες, και πως μετά την επικύρωση των αντιστοιχίσεων μπορεί να πραγματοποιηθεί απάντηση ερωτημάτων.

Η διατριβή αυτή οργανώνεται ως εξής:

- Στο κεφάλαιο 2 παρουσιάζουμε το μαθηματικό υπόβαθρο που είναι απαραίτητο για την κατανόηση του υπόλοιπου της διατριβής. Πιο συγκεκριμένα, αρχικά παρουσιάζουμε μια σύντομη περιγραφή των υπαρξιακών κανόνων με τους οποίους μπορούμε να εκφράσουμε ΠΛ που είναι επανεγγράψιμες. Συνεχίζοντας, κάνουμε μια σύντομη εισαγωγή στους κανόνες συμπερασμού ανάλυσης η χρήση των οποίων δίνει τη δυνατότητα για εξαγωγή νέας γνώσης από την ήδη υπάρχουσα. Στη συνέχεια, εισάγουμε τα Συζευκτικά Ερωτήματα, ενώ ακολούθως παρουσιάζουμε μια σύντομη εισαγωγή στην οικογένεια των Περιγραφικών Λογικών DL-Lite καθώς και στα προβλήματα συλλογιστικής των Περιγραφικών Λογικών γενικότερα. Τέλος, ορίζουμε το πρόβλημα της απάντησης ερωτημάτων μέσω επανεγγραφής, ενώ παρουσιάζουμε και τον αλγόριθμο επανεγγραφής ερωτημάτων PerfectRef [31].

- Στο κεφάλαιο 3 μελετάμε το πρόβλημα του υπολογισμού μιας επανεγγραφής για ερωτήματα που έχουν τροποποιηθεί, αποφεύγοντας να την υπολογίσουμε εξαρχής, χρησιμοποιώντας όσο το δυνατόν περισσότερη πληροφορία από την επανεγγραφή που έχει υπολογιστεί προηγουμένως για το αρχικό ερώτημα. Παρουσιάζουμε μια σειρά από παραδείγματα που μας καθοδηγούν διαισθητικά για την κατασκευή αλγορίθμων για ερωτήματα που έχουν τροποποιηθεί με προσθήκη ή αφαίρεση διακεκριμένων μεταβλητών ή ατόμων. Τέλος, παρουσιάζουμε μια σειρά από ιδιότητες που βελτιστοποιούν την απόδοση των αλγορίθμων μας, όπου αυτό κρίνεται απαραίτητο, και αφορούν στην αναγνώριση (μη-)περιττών και μη-υπαγόντων ερωτημάτων.

- Στο κεφάλαιο 4 εστιάζουμε στο πρόβλημα της επέκτασης ενός ερωτήματος με άτομα για την ΠΛ DL-Lite$_R$. Πιο συγκεκριμένα, μελετάμε το πρόβλημα υπολογισμού μιας επανεγγραφής για ερωτήματα που έχουν επεκταθεί με νέα άτομα,

δεδομένης μιας επανεγγραφής για το αρχικό ερώτημα, και παρουσιάζουμε μια σειρά από παραδείγματα που μας καθοδηγούν διαισθητικά στον αλγόριθμο μας. Επίσης παρουσιάζουμε αναλυτικούς αλγόριθμους και τέλος πώς μπορούμε να βελτιστοποιήσουμε το βήμα μείωσης που εμφανίζεται στον πρωτότυπο αλγόριθμο PerfectRef που προτάθηκε για την DL-Lite.

- Στο κεφάλαιο 5 μελετάμε το πρόβλημα της επαυξητικής επανεγγραφής ερωτημάτων για την ΠΛ DL-Lite$_R$ που στηρίζεται στην σταδιακή επεξεργασία των ατόμων ενός ερωτήματος και παρουσιάζουμε αναλυτικούς αλγόριθμους. Επίσης παρουσιάζουμε μια σειρά από βελτιστοποιήσεις που αφορούν στο βήμα κατά το οποίο επεξεργάζεται το τελευταίο άτομο του ερωτήματος και στην αναγνώριση (μη-)περιττών και μη-υπαγόντων ερωτημάτων. Οι βελτιστοποιήσεις αυτές όπως θα φανεί και στο επόμενο κεφάλαιο αυξάνουν σημαντικά την αποδοτικότητα του αλγορίθμου μας.

- Στο κεφάλαιο 6 παρέχουμε μια αναλυτική αξιολόγηση των προτεινόμενων αλγορίθμων, αντιπαραβάλλοντάς τους με υπάρχοντα συστήματα. Αρχικά, συγκρίνουμε τους αλγόριθμους επανεγγραφής τροποποιημένων ερωτημάτων που παρουσιάζουμε στο κεφάλαιο 3 χρησιμοποιώντας ένα πλαίσιο που αφορά σε οντολογίες DL-Lite$_R$ και ένα πλαίσιο που αφορά σε οντολογίες $\mathcal{ELHI}$. Τα αποτελέσματα μας δείχνουν πως οι αλγόριθμοι μας παρουσιάζουν εφάμιλη, και σε μερικές περιπτώσεις καλύτερη απόδοση από τα πιο γρήγορα συστήματα για επανεγγραφή ερωτημάτων για οντολογίες DL-Lite$_R$ και $\mathcal{ELHI}$, αντίστοιχα. Στη συνέχεια, όσον αφορά στους αλγορίθμους που σχετίζονται με την επαυξητική επανεγγραφή σταθερών ερωτημάτων για οντολογίες εκφραστικότητας DL-Lite$_R$, που περιγράφουμε στα κεφάλαια 4 και 5, συγκρίνουμε αρχικά τον πρωτότυπο αλγόριθμο PerfectRef με μια έκδοσή του η οποία χρησιμοποιεί το βελτιστοποιημένο βήμα μείωσης. Επίσης συγκρίνουμε τον αλγόριθμο PerfectRef (με το βελτιστοποιημένο βήμα μείωσης) με διάφορες εκδόσεις του δικού μας συστήματος. Τέλος, συγκρίνουμε το σύστημα μας με άλλα υπάρχοντα συστήματα επανεγγραφής ερωτημάτων που υποστηρίζουν την ίδια (ή και μεγαλύτερη) εκφραστικότητα και ήταν ελεύθερα διαθέσιμα. Τα αποτελέσματα μας δείχνουν ότι ο επαυξητικός υπολογισμός της επανεγγραφής είναι στην πλειονότητα των περιπτώσεων η πιο αποδοτική μέθοδος για την ΠΛ DL-Lite$_R$. Ειδικά, όταν τον συγκρίνουμε με τον αρχικό αλγόριθμο για την DL-Lite$_R$ [31], στον οποίο στηρίζεται ο αλγόριθμος μας, τα αποτελέσματα δείχνουν ότι η προσέγγιση μας είναι πιο γρήγορη για αρκετές τάξεις μεγέθους. Κάτι τέτοιο μπορεί να δικαιολογηθεί από τη πιο στοχευμένη στρατηγική που επεξεργάζεται ένα άτομο τη φορά σε αντίθεση με την ωμή προσέγγιση που ακολουθείται από τα περισσότερα συστήματα.

- Στο κεφάλαιο 7 μελετάμε το πρόβλημα της απάντησης ερωτημάτων πάνω από ένα δίκτυο οντολογιών. Για την κατασκευή ενός δικτύου οντολογιών είναι απαραίτητη η χρήση κάποιου αλγόριθμου αντιστοίχισης οντολογιών. Οι περισσότεροι όμως αλγόριθμοι αντιστοίχισης δεν λαμβάνουν υπ' όψιν τη σημασιολογία

των οντολογιών παράγοντας αντιστοιχίσεις που οδηγούν τις οντολογίες σε ασυνέπειες. Για το λόγο αυτό στο κεφάλαιο αυτό παρουσιάζουμε έναν αλγόριθμο επικύρωσης αντιστοιχίσεων που στηρίζει τη λειτουργία του στην ερμηνεία των αντιστοιχίσεων με χρήση της Ασαφούς Συνολοθεωρίας και της Ασαφούς Λογικής. Τέλος, παρουσιάζουμε τον αλγόριθμο μας για την απάντηση ερωτημάτων μέσω αντιστοιχίσεων.

- Στο κεφάλαιο 8 παρουσιάζουμε με μεγαλύτερη λεπτομέρεια βιβλιογραφία σχετική με τα προβλήματα που πραγματευόμαστε, ενώ το κεφάλαιο 9 κλείνει τη διατριβή συζητώντας πάλι τη συνεισφορά μας και παρουσιάζοντας θέματα για μελλοντική έρευνα.

Κεφάλαιο 2

Θεωρητικό Υπόβαθρο

Στο κεφάλαιο αυτό κάνουμε μια σύντομη εισαγωγή στην απαραίτητη ορολογία και παρουσιάζουμε θεωρίες, ορισμούς και αποτελέσματα που είναι απαραίτητα για την κατανόηση του υπολοίπου της διατριβής. Πιο συγκεκριμένα στην ενότητα 2.1 παρουσιάζουμε κάποιους βασικούς ορισμούς από τη Λογική Πρώτης Τάξης και κάνουμε μια εισαγωγή στους υπαρξιακούς κανόνες που μπορούν να χρησιμοποιηθούν για την αναπαράσταση οντολογιών. Συνεχίζοντας στην ενότητα 2.2 παρουσιάζουμε τον κανόνα συμπερασμού ανάλυσης ο οποίος χρησιμοποιείται για την εξαγωγή συμπερασμάτων από δοσμένες υποθέσεις. Στη συνέχεια, στην ενότητα 2.3 κάνουμε μια εισαγωγή στα Συζευκτικά Ερωτήματα παρουσιάζοντας τη σύνταξη και τη σημασιολογία τους. Ακολούθως, στην ενότητα 2.4 παρουσιάζουμε τη σύνταξη, τη σημασιολογία και τις υπηρεσίες συλλογιστικής της οικογένειας γλωσσων DL-Lite. Παρουσιάζουμε τις γλώσσες DL-Lite$_{core}$, DL-Lite$_F$ και την DL-Lite$_R$ καθώς και κάποια από τα προβλήματα συλλογιστικής που υποστηρίζουν οι ΠΛ (ενότητα 2.4.2). Επιπλέον παρουσιάζουμε το πρόβλημα απάντησης συζευκτικών ερωτημάτων μέσω της διαδικασίας της επανεγγραφής (ενότητα 2.5) και στην ενότητα 2.5.1 παρουσιάζουμε μια ανασκόπηση του αλγόριθμου επανεγγραφής ερωτημάτων PerfectRef ο οποίος μπορεί να εφαρμοστεί σε γλώσσες της οικογένειας DL-Lite. Τέλος στην ενότητα 2.6 παρουσιάζουμε μια μικρή εισαγωγή στους κατευθυνόμενους γράφους.

2.1 Υπαρξιακοί Κανόνες και Προτάσεις

Στην ενότητα αυτή παρουσιάζουμε κάποιους βασικούς ορισμούς από τη Λογική Πρώτης Τάξης που θα μας βοηθήσουν στον κατανόηση του υπολοίπου της διατριβής. Για πιο εκτενή ανάλυση ο ενδιαφερόμενος αναγνώστης παραπέμπεται στα [148, 53, 4, 123]. Έστω $\mathcal{P}$ πεπερασμένο ή μετρήσιμο σύνολο από *κατηγορήματα* (*predicates*), $\mathcal{F}$ περεπασμένο ή μετρήσιμο σύνολο από *συναρτησιακά σύμβολα* (*function symbols*), $\mathcal{C}$ πεπερασμένο ή μετρήσιμο σύνολο από *σταθερές* (*constants*), και $\mathcal{V}$ άπειρο σύνολο από *μεταβλητές* (*variables*). Κάθε κατηγόρημα σχετίζεται με ένα θετικό ακέραιο n που ονομάζεται *βαθμός του κατηγορήματος* (*arity*). Με $\Sigma(\mathcal{P}, \mathcal{F}, \mathcal{C}, \mathcal{V})$ δηλώνουμε τη γλώσσα Λογικής Πρώτης Τάξης που ορίζεται από τα $\mathcal{P}$, $\mathcal{F}$, $\mathcal{C}$ και $\mathcal{V}$.

Το σύνολο των *όρων* (*terms*) $\mathcal{T}(\Sigma)$ είναι το μικρότερο σύνολο για το οποίο

ισχύει $\mathcal{C} \cup \mathcal{V} \subseteq \mathcal{T}(\Sigma)$ ή εάν $f \in \mathcal{F}$ έχει βαθμό n και $t_i \in \mathcal{T}(\Sigma)$ για $1 \leq i \leq n$, τότε $f(t_1, \ldots, t_n) \in \mathcal{T}(\Sigma)$. Όροι της μορφής $f(t_1, \ldots, t_n)$ ονομάζονται *συναρτησιακοί όροι* (*functional terms*). Για τη συνέχεια της διατριβής θεωρούμε μόνο συναρτησιακούς όρους που έχουν βαθμό 1. Ένας όρος είναι *βασικός* (*ground term*) εάν δεν περιέχει μεταβλητές. Μια *ατομική φόρμουλα* (*atomic formula*) ή αλλιώς *άτομο* (*atom*) είναι μια έκφραση της μορφής $P(t_1, \ldots, t_n)$, όπου το P είναι ένα κατηγόρημα βαθμού n και $t_i \in \mathcal{T}(\Sigma)$ για $1 \leq i \leq n$ είναι όροι. Το σύνολο των *ατόμων* $\mathcal{A}(\Sigma)$ είναι το μικρότερο σύνολο για το οποίο εάν $P \in \mathcal{P}$ τότε $P(t_1, \ldots, t_n) \in \mathcal{A}(\Sigma)$. Για ένα πεπερασμένο σύνολο από άτομα $\{P_1, \ldots, P_n\}$, ορίζουμε με $\bigwedge\{P_1, \ldots, P_n\}$ τη φόρμουλα $P_1 \wedge \ldots \wedge P_n$. Επίσης, ένα *λεκτικό* (*literal*) L ειναι ένα άτομο ή η άρνηση ενός ατόμου. Στην πρώτη περίπτωση έχουμε ένα θετικό λεκτικό ενώ στη δεύτερη ένα αρνητικό λεκτικό.

Μια *αντικατάσταση* (*substitution*) είναι μια συνάρτηση με πεδίο ορισμού το $\mathcal{V}$ και πεδίο τιμών το $\mathcal{T}(\Sigma)$ η οποία μπορεί να αναπαρασταθεί ως το σύνολο των αντιστοιχίσεων $\{x_1 \mapsto t_1, \ldots, x_n \mapsto t_n\}$, όπου $x_i \in \mathcal{V}$ και $t_i \in \mathcal{T}(\Sigma)$. Η εφαρμογή μιας αντικατάστασης σ σε έναν όρο t, γράφεται $t\sigma$ και ορίζεται με $c\sigma = c$ εάν $c \in \mathcal{C}$, και $x\sigma = \sigma(x)$ εάν $x \in \mathcal{V}$. Η εφαρμογή μιας αντικατάστασης σ σε ένα άτομο $P(t_1, \ldots, t_n)$ ορίζεται ως $P(t_1\sigma, \ldots, t_n\sigma)$. Η *σύνθεση των αντικαταστάσεων* (*composition of substitutions*) δ και σ, γράφεται ως $\delta \circ \sigma$ ή $\delta\sigma$, και για κάποιο όρο t ορίζεται ως $t\delta \circ \sigma = t\delta\sigma = (t\delta)\sigma$. Μια αντικατάσταση σ είναι *πιο γενική* (*more general*) από μια αντικατάσταση δ εάν υπάρχει μια αντικατάσταση λ τ.ω. $\delta = \sigma\lambda$. Επιπλέον, μια αντικατάσταση σ ονομάζεται *ενοποιητής* (*unifier*) των όρων s και t εάν ισχύει $s\sigma = t\sigma$ ενώ εάν η σ είναι πιο γενική από κάθε άλλο ενοποιητή (για τα άτομα αυτά) τότε ονομάζεται *πιο γενικός ενοποιητής* (*most general unifier – mgu*). Επιπλέον η αντικατάσταση $\sigma = \{\}$ ονομάζεται *ταυτοτική αντικατάσταση* (*identity substitution*). Επιπρόσθετα, κάθε αντικατάσταση σ επάγει ένα κατευθυνόμενο γράφο $G = \langle V, E \rangle$, όπου $t \in V$ ανν ο t είναι ένας όρος στην σ και $\langle x, t \rangle \in E$ ανν $x \mapsto t \in \sigma$.[1] Τέλος με $dom(\sigma) = \{x_1, \ldots, x_n\}$ αναπαριστούμε το σύνολο μεταβλητών του πεδίου ορισμού της αντικατάστασης.

Μια *πρόταση* (*clause*) είναι μια διάζευξη λεκτικών

$$L_1 \vee \ldots \vee L_n$$

Μια πρόταση θα λέγεται *πρόταση Horn* (*Horn clause*) εάν το πολύ ένα από τα λεκτικά είναι θετικό. Για παράδειγμα η πρόταση

$$\neg\text{παρακολουθεί}(x, y) \vee \neg\text{Μεταπτυχιακό}(y) \vee \text{ΜεταπτυχιακόςΦοιτητής}(x)$$

είναι μια πρόταση Horn, όπου τα $\neg$παρακολουθεί(x, y) και $\neg$Μεταπτυχιακό(y) είναι αρνητικά λεκτικά για τα κατηγορήματα παραλοκουθεί και Μεταπτυχιακό αντίστοιχα, και το ΜεταπτυχιακόςΦοιτητής(x) είναι θετικό λεκτικό για το κατηγόρημα ΜεταπτυχιακόςΦοιτητής. Η πρόταση αυτή γράφεται ισοδύναμα και ως:

$$\text{παρακολουθεί}(x, y) \wedge \text{Μεταπτυχιακό}(y) \rightarrow \text{ΜεταπτυχιακόςΦοιτητής}(x)$$

που δηλώνει πως εάν κάποιος παρακολουθεί κάποιο μεταπτυχιακό μάθημα, τότε είναι μεταπτυχιακός φοιτητής. Σε μια πρόταση Horn θεωρούμε πως οι ελεύθερες μεταβλητές των λεκτικών προέρχονται από καθολικούς ποσοδείκτες που εμφανίζονται στην αρχή του τύπου.

[1] Μια μικρή εισαγωγή στους κατευθυνόμενους γράφους δίνεται στην ενότητα 2.6.

Ένας *υπαρξιακός κανόνας* (*existential rule*) (ή απλά κανόνας) [11, 26], που συχνά ονομάζεται και *αξίωμα*, είναι μια φόρμουλα της μορφής

$$\forall \vec{x}.\forall \vec{z}.[\phi(\vec{x},\vec{z}) \rightarrow \exists \vec{y}.\psi(\vec{x},\vec{y})]$$

όπου $\phi(\vec{x},\vec{z})$ και $\psi(\vec{x},\vec{y})$ είναι συζεύξεις από άτομα που δεν περιέχουν συναρτησιακούς όρους και τα $\vec{x},\vec{y}$ και $\vec{z}$ είναι ανά δύο διαφορετικά. Η φόρμουλα ϕ ονομάζεται *σώμα*, η φόρμουλα ψ ονομάζεται *κεφαλή* και οι καθολικοί ποσοδείκτες συχνά παραλείπονται. Για παράδειγμα οι φόρμουλες

$$\textsf{Καθηγητής}(x) \rightarrow \exists y.\textsf{διδάσκει}(x,y) \wedge \textsf{Μάθημα}(y) \tag{2.1}$$

$$\textsf{ΠροπτυχιακόςΦοιτητής}(x) \rightarrow \textsf{Φοιτητής}(x) \tag{2.2}$$

είναι υπαρξιακοί κανόνες. Ο πρώτος δηλώνει πως εάν κάποιος είναι καθηγητής τότε υπάρχει κάποιο μάθημα το οποίο διδάσκει, ενώ ο δεύτερος πως εάν κάποιος είναι προπτυχιακός φοιτητής τότε θα είναι και φοιτητής γενικότερα. Όπως φαίνεται και από το παράδειγμα οι καθολικές μεταβλητές παραλείπονται χωρίς βλάβη της γενικότητας. Όταν το διάνυσμα $\vec{y}$ είναι κενό τότε ο κανόνας ονομάζεται *κανόνας Datalog* (*Datalog rule*). Για ένα κανόνα Datalog r, ορίζουμε με $\mathsf{bd}(r)$ το σύνολο των ατόμων του σώματος του r, και με $\mathsf{hd}(r)$ τα άτομα της κεφαλής του. Ένα *πρόγραμμα Datalog* (*Datalog program*) είναι ένα πεπερασμένο σύνολο από κανόνες Datalog. Αξίζει να σημειωθεί ότι εξ' ορισμού οι κανόνες Datalog είναι *ασφαλείς* (*safe*)—δηλαδή, όλες οι μεταβλητές που εμφανίζονται στο διάνυσμα $\vec{x}$ της κεφαλής του εμφανίζονται και στο σώμα του.

Ένας υπαρξιακός κανόνας μπορεί να μετατραπεί σε μια πρόταση (Horn) εάν μετατραπεί σε *Κανονική Μορφή Skolem* (*Skolem Normal Form*) [147, 74] με την αφαίρεση των υπαρξιακών ποσοδεικτών και την εισαγωγή νέων συναρτησιακών συμβόλων. Για παράδειγμα ο υπαρξιακός κανόνας (2.1) μπορεί να εκφραστεί με τις παρακάτω προτάσεις

$$\textsf{Καθηγητής}(x) \rightarrow \textsf{διδάσκει}(x, f(x))$$
$$\textsf{Καθηγητής}(x) \rightarrow \textsf{Μάθημα}(f(x))$$

στις οποίες αντί της μεταβλητής y έχουμε τον (ίδιο) συναρτησιακό όρο $f(x)$, ενώ ο υπαρξιακός κανόνας (2.2) μπορεί να εκφραστεί με την πρόταση

$$\textsf{ΠροπτυχιακόςΦοιτητής}(x) \rightarrow \textsf{Φοιτητής}(x)$$

Πάρα πολλές γλώσσες οντολογιών Horn, όπως είναι και η οικογένεια γλωσσών της DL-Lite [31] (που θα περιγραφεί στη συνέχεια), η $\mathcal{ELHI}$ [117], καθώς και η Datalog$^{\pm}$ [26] μπορούν να αναπαρασταθούν με τη χρήση υπαρξιακών κανόνων. Για το λόγο αυτό στη συνέχεια της διατριβής μας οι υπαρξιακοί κανόνες πολλές φορές θα χρησιμοποιούνται αντί των αξιωμάτων υπαγωγής για τον ορισμό μιας οντολογίας.

2.2 Κανόνας Συμπερασμού Ανάλυσης

Ο *λογισμός της ανάλυσης* (*resolution calculus*) [125], είναι μια μέθοδος που χρησιμοποιείται ευρέως στην *απόδειξη θεωρημάτων* (*theorem proving*). Σύμφωνα με αυτόν

από κάθε σύνολο μη-ικανοποιήσιμων προτάσεων μπορεί να παραχθεί μια αντίφαση. Ο λογισμός της ανάλυσης στηρίζεται σε δύο *κανόνες συμπερασμού* (*inference rules*). Τον *κανόνα της ανάλυσης* (*resolution*)

$$\frac{C \vee A \quad D \vee \neg B}{(C \vee D)\sigma}$$

και τον *κανόνα της παραγοντοποίησης* (*factoring*)

$$\frac{C \vee A \vee B}{(C \vee A)\sigma}$$

όπου και στις δύο περιπτώσεις, σ είναι ένας πιο γενικός ενοποιητής των ατόμων A και B. Οι κανόνες αυτοί δοθέντων κάποιων υποθέσεων παρέχουν τη δυνατότητα εξαγωγής νέων προτάσεων (συμπερασμάτων) που ικανοποιούν τις υποθέσεις αυτές.

Για τους σκοπούς της παρούσας διατριβής ένας *κανόνας συμπερασμού ανάλυσης* (*resolution inference rule*) είναι μια σχέση ανάμεσα σε προτάσεις η οποία υποσημειώνεται με μια αντικατάσταση και μια λίστα από σύνολα ατόμων. Τα στοιχεία της σχέσης αυτής μαζί με την αντίστοιχη αντικατάσταση και τις λίστες γράφονται ως εξής:

$$\frac{C \quad C_1 \ldots C_n}{C'}[\sigma, \Upsilon]$$

όπου C είναι μια πρόταση που ονομάζεται *κύρια υπόθεση* (*main premise*), $C_1, \ldots, C_n$ είναι διακριτές προτάσεις που ονομάζονται *παράπλευρες υποθέσεις* (*side premises*), και C' είναι το *συμπέρασμα* (*conclusion*). Ένας συμπερασμός ορίζεται επίσης από μια πλειάδα της μορφής $\langle C, C_1, \ldots, C_n, C' \rangle$. Για ένα συμπερασμό $\langle C, C_1, \ldots, C_n, C' \rangle$, το C' απαιτείται να είναι ένα αναλυθέν της C με τις $C_1, \ldots, C_n$ μέσω της σύνθεσης του πιο γενικού ενοποιητή θ και μιας μετονομασίας ρ. Το γεγονός αυτό εξασφαλίζει ουσιαστικά ότι το C' δεν μοιράζεται μεταβλητές με καμία από τις υποθέσεις. Επιπρόσθετα, σ είναι μια αντιστοίχιση από τις μεταβλητές της C σε αυτές του C' μέσω της οποίας θεσπίζεται μια σύνδεση μεταξύ τους. Πιο συγκεκριμένα, ορίζεται ως το υποσύνολο των αναθέσεων στο $\theta \circ \rho$ που αντιστοιχίζει μεταβλητές από την C σε μη-συναρτησιακούς όρους του C'. Επιπλέον, Υ είναι μια λίστα από ζεύγη από σύνολα της μορφής $\langle \mathcal{A}_i, \mathcal{B}_i \rangle$ που περιέχουν όρους σχετικούς με τον συμπερασμό. Πιο συγκεκριμένα, $\mathcal{A}_i$ είναι όλοι οι όροι από την C που εννοποιούνται με όρους από την i-οστή παράπλευρη υπόθεση C_i, ενώ $\mathcal{B}_i$ είναι οι όροι της C_i που εισάγονται στο συμπέρασμα C' (μετά την εφαρμογή της $\theta \circ \rho$). Τυπικά, $C' = [(C \setminus \bigcup_i \mathcal{A}_i)\theta \circ \rho \cup \bigcup_i \mathcal{B}_i]$, όπου $\mathcal{B}_i \subseteq C_i\theta \circ \rho$. Για παράδειγμα, εάν έχουμε τις προτάσεις

$$\begin{aligned} C &= \textsf{ΠροπτυχιακόςΦοιτητής}(x) \rightarrow \textsf{Φοιτητής}(x) \\ C_1 &= \textsf{παρακολουθεί}(x_1, y_1) \wedge \textsf{Προπτυχιακό}(y_1) \rightarrow \textsf{ΠροπτυχιακόςΦοιτητής}(x_1) \end{aligned}$$

τότε εφαρμόζοντας τον κανόνα συμπερασμού με κύρια υπόθεση την πρόταση C και παράπλευρη την C_1 παράγεται η πρόταση

$$C' = \textsf{παρακολουθεί}(x', y') \wedge \textsf{Προπτυχιακό}(y') \rightarrow \textsf{Φοιτητής}(x')$$

με $\sigma = \{x \mapsto x'\}$ και $\Upsilon = \{\langle \mathcal{A}, \mathcal{B} \rangle\}$, με $\mathcal{A} = \{\text{ΠροπτυχιακόςΦοιτητής}(x)\}$ και $\mathcal{B} = \{\text{παρακολουθεί}(x', y'), \text{Προπτυχιακό}(y')\}$.

Επιπρόσθετα ένας *κανόνας παραγοντοποίησης* (*factoring inference rule*) είναι ένας συμπερασμός για τον οποίο ισχύει $n = 0$, δηλαδή μια σχέση ανάμεσα σε δύο προτάσεις η οποία υποσημειώνεται με μια αντικατάσταση και μια λίστα από σύνολα ατόμων. Αυτή γράφεται ως εξής:

$$\frac{C}{C'}[\sigma, \Upsilon]$$

και ορίζεται επίσης από μια πλειάδα της μορφής $\langle C, C' \rangle$. Το σ ορίζεται όπως και πριν ενώ εφόσον $n = 0$ έχουμε $\Upsilon = \{\langle \emptyset, \emptyset \rangle\}$ με αποτέλεσμα το συμπέρασμα C' να δίνεται από τη σχέση $C' = [C\theta \circ \rho]$. Για παράδειγμα εάν έχουμε την πρόταση

$$C = \text{διδάσκει}(x, y) \wedge \text{διδάσκει}(z, y) \rightarrow \text{Καθηγητής}(x)$$

τότε εφαρμόζοντας τον κανόνα παραγοντοποίησης παράγεται η πρόταση

$$C' = \text{διδάσκει}(x', y') \rightarrow \text{Καθηγητής}(x')$$

με $\sigma = \{x \mapsto x', z \mapsto x'\}$.

Ένα σύστημα συμπερασμού Γ είναι μια συλλογή από κανόνες συμπερασμού για ένα σύνολο από προτάσεις N. Για ένα σύνολο προτάσεων N με $\Gamma(N)$ ορίζουμε το σύνολο όλων των συμπερασμών που προκύπτουν από το σύστημα Γ και έχουν όλες τις υποθέσεις τους στο N.

2.3 Συζευκτικά Ερωτήματα

Στην ενότητα αυτή κάνουμε μια εισαγωγή στη σύνταξη και σημασιολογία των συζευκτικών ερωτημάτων (ΣΕ) (Conjunctive Queries – CQ). Ένα *ερώτημα* (*query*) $\mathcal{Q}$ είναι ένα πεπερασμένο σύνολο από κανόνες Datalog (πρόγραμμα Datalog) που περιέχουν ένα διακριτό κατηγόρημα ερωτήματος $\mathcal{Q}_A$ στο άτομο κεφαλής. Ένα ερώτημα $\mathcal{Q}$ αποτελεί ένα *Σύνολο από Συζευκτικά Ερωτήματα (ΣΣΕ)* (*Union of Conjunctive Queries – UCQ*) εάν είναι ένα πρόγραμμα Datalog όλοι οι κανόνες του οποίου περιέχουν το κατηγόρημα $\mathcal{Q}_A$ στην κεφαλή αλλά όχι στο σώμα τους. Ένα ΣΣΕ $\mathcal{Q}$ ονομάζεται *Συζευκτικό Ερώτημα (ΣΕ)* (*Conjunctive Query – CQ*) εάν αποτελείται από ακριβώς έναν κανόνα. Στη συνέχεια της διατριβής μας κάνουμε συχνά κατάχρηση του συμβολισμού και χρησιμοποιούμε το $\mathcal{Q}$ για να αναφερθούμε στον μοναδικό κανόνα ενός ΣΕ.

Όλες οι μεταβλητές που εμφανίζονται στην κεφαλή ενός ΣΕ ονομάζονται *διακεκριμένες* ή *μεταβλητές απάντησης* (*distinguished* ή *answer variables*). Όλες οι άλλες μεταβλητές του ερωτήματος ονομάζονται *μη-διακεκριμένες* (*undistinguished*). Για ένα ερώτημα $\mathcal{Q}$, $\text{var}(\mathcal{Q})$ είναι το σύνολο όλων των μεταβλητών που εμφανίζονται στο $\mathcal{Q}$ και $\text{avar}(\mathcal{Q})$ το σύνολο όλων των διακεκριμένων μεταβλητών του. Μια μεταβλητή που είναι είτε διακεκριμένη ή εμφανίζεται στο $\mathcal{Q}$ τουλάχιστον δύο φορές ονομάζεται *δεσμευμένη* (*bound*), αλλιώς ονομάζεται *μη-δεσμευμένη* (*unbound*). Για παράδειγμα στο ερώτημα

$$\mathcal{Q}_1 = \text{Φοιτητής}(x) \wedge \text{συμφοιτητήςΜε}(x, y) \rightarrow \mathcal{Q}_A(x)$$

η μεταβλητή x είναι διακεκριμένη (και συνεπώς δεσμευμένη) ενώ η μεταβλητή y είναι μη-διακεκριμένη και επειδή εμφανίζεται μόνο μια φορά στο σώμα του ερωτήματος μη-δεσμευμένη. Από την άλλη στο ερώτημα

$$\mathcal{Q}_2 = \textsf{Φοιτητής}(x) \wedge \textsf{συμφοιτητήςΜε}(x, y) \wedge \textsf{παρακολουθεί}(y, z) \rightarrow \mathcal{Q}_A(x)$$

η μεταβλητή x είναι διακεκριμένη και άρα δεσμευμένη όμοια με πριν, η μεταβλητή y είναι μη-διακεκριμένη και δεσμευμένη, εφόσον εμφανίζεται δύο φορές στο σώμα του ερωτήματος και τέλος, η μεταβλητή z είναι μη-διακεκριμένη και μη-δεσμευμένη, αφού εμφανίζεται μόνο μια φορά.

Έστω ένα ΣΕ $\phi(\vec{x}, \vec{z}) \rightarrow Q_A(\vec{x})$, όπου $\vec{x} = (x_1, \ldots, x_n)$ και επίσης έστω ένα διάνυσμα μεταβλητών $\vec{y} = (y_1, \ldots, y_m)$. Τότε με την έκφραση $\phi(\vec{x}, \vec{z}) \rightarrow Q_A(\vec{x}, \vec{y})$ θεωρούμε το νέο ερώτημα $\phi(\vec{x}, \vec{z}) \rightarrow Q_A(\vec{k})$, όπου $\vec{k} = (x_1, \ldots, x_n, y_1, \ldots, y_m)$. Έστω τώρα ένα ΣΕ $\mathcal{Q} = \phi(\vec{x}, \vec{z}) \rightarrow Q_A(\vec{x})$, όπου $\vec{x} = (x_1, \ldots, x_n)$. Έστω επίσης μια ακολουθία από θετικούς ακεραίους $j_1, \ldots, j_m$ για την οποία $n \geq \max\{j_1, \ldots, j_m\}$. Η *προβολή* (*projection*) ενός ΣΕ $\mathcal{Q}$ στην ακολουθία $j_1, \ldots, j_m$, γράφεται $\pi_{j_1,\ldots,j_m}(\mathcal{Q})$, και είναι το νέο ΣΕ $\phi(\vec{x}, \vec{z}) \rightarrow Q_A(\vec{x}')$ όπου $\vec{x}' = (x_{j_1}, \ldots, x_{j_m})$. Η σχέση $\pi_{j_1,\ldots,j_m}$ ονομάζεται *τελεστής προβολής* (*projection operator*). Εφαρμόζοντας για παράδειγμα τον τελεστή προβολής π_2 στο ερώτημα

$$\mathcal{Q}_3 = \textsf{συμφοιτητήςΜε}(x, y) \rightarrow \mathcal{Q}_A(x, y)$$

παίρνουμε το ερώτημα

$$\pi_2(\mathcal{Q}_3) = \textsf{συμφοιτητήςΜε}(x, y) \rightarrow \mathcal{Q}_A(y).$$

Δεδομένων δύο ΣΕ $\mathcal{Q}$, $\mathcal{Q}'$ με διακεκριμένες μεταβλητές $\vec{x}$ και $\vec{y}$ αντίστοιχα, λέμε ότι το $\mathcal{Q}$ *υπάγει* (*subsumes*) το $\mathcal{Q}'$ [38, 1], εάν υπάρχει αντικατάσταση σ από τις μεταβλητές του $\mathcal{Q}$ στις μεταβλητές του $\mathcal{Q}'$ τ.ω. το σύνολο $\mathcal{Q}\sigma$ να είναι υποσύνολο του $\mathcal{Q}'$, δηλ

$$\mathcal{Q}\sigma \subseteq \mathcal{Q}'.$$

Για παράδειγμα το ερώτημα $\mathcal{Q}_1$ που ορίσαμε παραπάνω υπάγει το ερώτημα $\mathcal{Q}_2$ με την ταυτοτική αντικατάσταση. Επιπλέον θα λέμε ότι δύο ερωτήματα $\mathcal{Q}$ και $\mathcal{Q}'$ είναι *ισοδύναμα* (*equivalent*) εάν το $\mathcal{Q}$ υπάγει το $\mathcal{Q}'$ και το $\mathcal{Q}'$ υπάγει το $\mathcal{Q}$.

Από τις ιδιότητες της σχέσης «υπάγει» [38] έχουμε ότι εάν το ερώτημα $\mathcal{Q}$ υπάγει το $\mathcal{Q}'$ τότε μπορεί να συμβαίνουν δύο τινά. Είτε το $\mathcal{Q}$ υπάγει ένα ερώτημα που προκύπτει από την εφαρμογή ενός κανόνα συμπερασμού στο $\mathcal{Q}'$, έστω $\mathcal{Q}''$, είτε το ερώτημα που προκύπτει από την εφαρμογή ενός κανόνα συμπερασμού στο $\mathcal{Q}$ υπάγει το $\mathcal{Q}''$. Επιπλέον, για ένα ΣΣΕ $\mathcal{Q}$ και ένα ΣΕ $\mathcal{Q}_1$, λέμε ότι το $\mathcal{Q}_1$ είναι *περιττό* (*redundant*) στο $\mathcal{Q}$ εάν υπάρχει κάποιο ερώτημα $\mathcal{Q}_2 \in \mathcal{Q}$ τ.ω. $\mathcal{Q}_2 \neq \mathcal{Q}_1$ και το $\mathcal{Q}_2$ υπάγει το $\mathcal{Q}_1$. Διαφορετικά λέμε ότι το $\mathcal{Q}_1$ είναι *μη-περιττό* (*non-redundant*) στο $\mathcal{Q}$. Ακόμα, για ένα ΣΣΕ $\mathcal{Q}$ και ένα ΣΕ $\mathcal{Q}_1$, λέμε ότι το $\mathcal{Q}_1$ είναι *μη-υπάγον* (*non-subsumer*) στο $\mathcal{Q}$ εάν δεν υπάρχει κάποιο ερώτημα $\mathcal{Q}_2 \in \mathcal{Q}$ τ.ω. $\mathcal{Q}_2 \neq \mathcal{Q}_1$ και το $\mathcal{Q}_1$ υπάγει το $\mathcal{Q}_2$. Επιπλέον λέμε ότι το $\mathcal{Q}_1$ είναι *ισοδύναμο* (*equivalent*) με το $\mathcal{Q}_2$ εάν το ένα υπάγει το άλλο και αντίστροφα. Τέλος, για ένα ΣΣΕ $\mathcal{Q}$ λέμε ότι είναι *μη-περιττό ΣΣΕ* (*non-redundant UCQ*) εάν δεν υπάρχουν $\mathcal{Q}_1, \mathcal{Q}_2 \in \mathcal{Q}$ τ.ω. $\mathcal{Q}_1 \neq \mathcal{Q}_2$ και το $\mathcal{Q}_1$ υπάγει το

Αλγόριθμος 1 removeRedundant($\mathcal{Q}$)

Είσοδος: Ένα ΣΣΕ $\mathcal{Q}$.

1: Αρχικοποίησε ένα ΣΣΕ $\mathcal{Q}' = \mathcal{Q}$
2: **for all** $\mathcal{Q}_1 \in \mathcal{Q}$ **do**
3: **for all** $\mathcal{Q}_2 \in \mathcal{Q}$ με $\mathcal{Q}_1 \neq \mathcal{Q}_2$ **do**
4: **if** το $\mathcal{Q}_1$ υπάγει το $\mathcal{Q}_2$ και $\mathcal{Q}_1 \in \mathcal{Q}'$ **then**
5: Αφαίρεσε το $\mathcal{Q}_2$ από το $\mathcal{Q}'$
6: **end if**
7: **end for**
8: **end for**
9: **return** $\mathcal{Q}'$

$\mathcal{Q}_2$. Για να μπορέσουμε να υπολογίσουμε ένα μη-περιττό ΣΣΕ χρησιμοποιούμε τον αλγόριθμο απαλοιφής περιττών ερωτημάτων removeRudantant [116] που φαίνεται στον Αλγόριθμο 1. Σύμφωνα με τον αλγόριθμο αυτό για να πάρουμε το μη-περιττό ΣΣΕ εξετάζουμε όλα τα ερωτήματα του ΣΣΕ εισόδου (ανά ζεύγη) και όταν ένα ερώτημα $\mathcal{Q}_1$ υπάγεται από κάποιο άλλο $\mathcal{Q}_2$ τότε το αφαιρούμε από το ΣΣΕ.

Για τη συνέχεια της διατριβής μας θεωρούμε ότι όλα τα ΣΕ είναι *συνδεδεμένα* (*connected*) [58]. Πιο συγκεκριμένα, έστω ΣΕ $\mathcal{Q}$. Λέμε ότι το $\mathcal{Q}$ είναι συνδεδεμένο εάν, για όλους τους όρους t, t', υπάρχει ακολουθία $t_1, \ldots, t_n$ τ.ω. $t_1 = t$, $t_n = t'$ και, για όλα τα $1 \leq i < n$, υπάρχει ρόλος R τ.ω. $R(\ldots, t_i, \ldots, t_{i+1}, \ldots) \in \mathcal{Q}$.

2.4 Η Οικογένεια Γλωσσών της DL-Lite

Οι Περιγραφικές Λογικές (ΠΛ) (Description Logics – DLs) [9] είναι μια οικογένεια γλωσσών αναπαράστασης γνώσης, που βασίζονται στη λογική (logic-based), και έχουν σχεδιαστεί έτσι ώστε η καταγραφή της γνώσης και η διενέργεια εργασιών συλλογιστικής πάνω σε αυτές να γίνεται με ένα δομημένο και κατανοητό τρόπο. Η καταγραφή της γνώσης στις ΠΛ γίνεται με τη χρήση *ατόμων* (*individuals*), *εννοιών* (*concepts*) που αναπαριστούν σύνολα από άτομα και *ρόλων* (*roles*) που αναπαριστούν δυαδικές σχέσεις ανάμεσα στα άτομα καθώς και με τη χρήση *κατασκευαστών* (*constructors*) που επενεργούν πάνω στις έννοιες και τους ρόλους κατασκευάζοντας άλλες πιο σύνθετες. Για παράδειγμα η σύνθετη έννοια

$$\text{Φοιτητής} \sqcap \exists \text{παρακολουθεί.Μάθημα}$$

αναπαριστά εκείνους τους φοιτητές που παρακολουθούν τουλάχιστον ένα μάθημα. Το σύνολο των κατασκευαστών που μπορούν να χρησιμοποιηθούν για την περιγραφή γνώσης είναι χαρακτηριστικό για κάθε ΠΛ. Στη συνέχεια με χρήση διαφόρων υπηρεσιών συλλογιστικής μπορούμε να εξάγουμε νέα, υπονοούμενη γνώση από την περιγραφείσα. Οι λόγοι για τους οποίους οι ΠΛ έχουν γίνει ιδιαίτερα γνωστές και χρησιμοποιούνται σε πάρα πολλές εφαρμογές είναι εκτός από το ότι είναι πολύ απλές στη χρήση, ακόμα και από μη ειδικούς, το γεγονός ότι είναι *εύρωστα αποφασίσιμες* (*robustly decidable*).

Ορισμός 2.4.1. Έστω **C**, **R** και **I** μετρήσιμα και ανά ζεύγη διαφορετικά σύνολα από ατομικές έννοιες, ατομικούς ρόλους, και άτομα.

Οι DL-Lite$_{core}$-ρόλοι κατασκευάζονται με βάση τον παρακάτω κανόνα σύνταξης, όπου R ένας ρόλος που λέγεται *βασικός* (*basic role*), E ένας ρόλος που λέγεται *γενικός* (*general role*) και $P \in \mathbf{R}$:

$$R ::= P \mid P^- \qquad E ::= R \mid \neg R$$

Ένας βασικός ρόλος της μορφής P^- ονομάζεται *αντίστροφος* (*inverse*) ρόλος του P.

Οι DL-Lite$_{core}$-έννοιες κατασκευάζονται με βάση τον παρακάτω κανόνα σύνταξης, όπου B μια *βασική έννοια* (*basic concept*), C μια *γενική έννοια* (*general concept*), $A \in \mathbf{C}$ και R ένας βασικός ρόλος:

$$B ::= A \mid \exists R.\top \qquad C ::= B \mid \neg B$$

$\triangle$

Η έννοια $\top$ ονομάζεται *καθολική* (*top*) έννοια, ενώ η έννοια $\exists R.\top$ (που από εδώ και στο εξής θα εμφανίζεται και ως $\exists R$) ονομάζεται *απροσδιόριστος υπαρξιακός περιορισμός* (*unqualified existential restriction*).

Ένα από τα χαρακτηριστικά που κάνει τις γλώσσες αναπαράστασης γνώσης να διαφέρουν από τις φυσικές γλώσσες είναι το γεγονός ότι είναι τυπικές. Αυτό σημαίνει ότι τα δομικά στοιχεία των γλωσσών αυτών ερμηνεύονται με ένα τυπικό, μαθηματικό τρόπο, έτσι ώστε το νόημα μιας δήλωσης να μπορεί να οριστεί αυστηρά/τυπικά, κάτι το οποίο δεν συμβαίνει με τις φυσικές γλώσσες, στις οποίες οι δηλώσεις μπορεί να εμπεριέχουν αμφισημία. Μια *ερμηνεία* (*interpretation*) Περιγραφικής Λογικής $\mathcal{I}$ ορίζεται από ένα ζεύγος $(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, όπου $\Delta^{\mathcal{I}}$ είναι ένα μη-κενό σύνολο που ονομάζεται *χώρος ερμηνείας* (*domain of interpretation*) που περιέχει στοιχεία που ονομάζονται *αντικείμενα* (*objects*), και $\cdot^{\mathcal{I}}$ είναι μια *συνάρτηση ερμηνείας* (*interpretation function*) που απεικονίζει:

- κάθε άτομο $a \in \mathbf{I}$ σε ένα αντικείμενο $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$,
- κάθε ατομική έννοια $A \in \mathbf{C}$ σε ένα υποσύνολο $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, και
- κάθε ατομικό ρόλο $P \in \mathbf{R}$ σε μια δυαδική σχέση $P^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$.

Τέλος η συνάρτηση ερμηνείας μπορεί να επεκταθεί για να δώσει ερμηνεία και στις σύνθετες έννοιες και ρόλους της DL-Lite$_{core}$. Οι ερμηνείες αυτές φαίνονται στον Πίνακα 2.1.

Βλέπουμε λοιπόν ότι κάθε DL-Lite$_{core}$-έννοια ερμηνεύεται ως ένα υποσύνολο του $\Delta^{\mathcal{I}}$. Για παράδειγμα, η έννοια $\top$ ερμηνεύεται ως το σύνολο το οποίο περιέχει όλα τα αντικείμενα του χώρου ερμηνείας. Επιπλέον, η ερμηνεία της έννοιας $\exists R$ περιέχει το σύνολο των αντικειμένων τα οποία συμμετέχουν στη σχέση $R^{\mathcal{I}}$ με κάποιο άλλο αντικείμενο του $\Delta^{\mathcal{I}}$. Για να έχει νόημα μια έννοια θα πρέπει να υπάρχει κάποια ερμηνεία η οποία να ικανοποιεί τα αξιώματα του $\mathcal{T}$, να είναι δηλαδή μοντέλο του $\mathcal{T}$, έτσι ώστε η έννοια να ερμηνεύεται σε ένα μη-κενό σύνολο. Μια έννοια που ικανοποιεί αυτή

Πίνακας 2.1: Σημασιολογία DL-Lite$_{core}$-εννοιών και ρόλων

Κατασκευαστής	**Σύνταξη**	**Σημασιολογία**
καθολική έννοια	$\top$	$\Delta^{\mathcal{I}}$
άρνηση έννοιας	$\neg B$	$\Delta^{\mathcal{I}} \setminus B^{\mathcal{I}}$
υπαρξιακός περιορισμός	$\exists R.\top$	$\{x \in \Delta^{\mathcal{I}} \mid \exists y.(x,y) \in R^{\mathcal{I}}\}$
αντίστροφοι ρόλοι	P^-	$\{\langle x,y\rangle \mid \langle y,x\rangle \in P^{\mathcal{I}}\}$
άρνηση ρόλου	$\neg R$	$\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \setminus R^{\mathcal{I}}$

την ιδιότητα λέγεται *ικανοποιήσιμη* (*satisfiable*) σε σχέση με το $\mathcal{T}$, ενώ διαφορετικά λέγεται *μη-ικανοποιήσιμη* (*unsatisfiable*).

Μέχρι στιγμής έχουμε δει πώς μπορούμε να χρησιμοποιούμε τους κατασκευαστές που μας προσφέρει η DL-Lite$_{core}$ σε συνδυασμό με τις ατομικές έννοιες και ρόλους με σκοπό τη δημιουργία σύνθετων εννοιών και ρόλων. Οι ΠΛ όμως μας προσφέρουν τη δυνατότητα, να μπορούμε να αποδίδουμε ονόματα στις σύνθετες έννοιες που θέλουμε να περιγράψουμε καθώς επίσης και να περιγράψουμε σχέσεις ανάμεσα σε αυτές. Οι σχέσεις αυτές παρουσιάζονται με τη μορφή αξιωμάτων που ονομάζονται *αξιώματα ορολογίας* (*terminological axioms*), συλλέγονται στο *σώμα ορολογίας* (*terminological box – TBox*), που συμβολίζεται με $\mathcal{T}$ και είναι ένα σύνολο από ισχυρισμούς της μορφής

$$B \sqsubseteq C$$

όπου B και C είναι DL-Lite$_{core}$-έννοιες, όπως αυτές έχουν οριστεί στον Ορισμό 2.4.1. Τα αξιώματα αυτά ονομάζονται και *αξιώματα υπαγωγής* (*subsumption axioms* ή *inclusion axioms*). Διαισθητικά ένα αξίωμα της μορφής $B \sqsubseteq C$ δηλώνει ότι η έννοια C είναι πιο γενική της έννοιας B ή αλλιώς ότι η έννοια B είναι υποέννοια της έννοιας C. Συνεπώς, εφόσον οι έννοιες B και C ερμηνεύονται σαν σύνολα είναι φυσικό να πούμε ότι μια ερμηνεία $\mathcal{I}$ *ικανοποιεί* (*satisfies*) ένα αξίωμα υπαγωγής $B \sqsubseteq C$ εάν $B^{\mathcal{I}} \subseteq C^{\mathcal{I}}$, δηλαδή εάν η ερμηνεία $\mathcal{I}$ ερμηνεύει την έννοια C ως υπερσύνολο της έννοιας B. Μια ερμηνεία $\mathcal{I}$ ικανοποιεί ένα σώμα ορολογίας $\mathcal{T}$ εάν ικανοποιεί όλα τα αξιώματα στο $\mathcal{T}$. Τότε λέμε ότι η $\mathcal{I}$ είναι *μοντέλο* (*model*) του $\mathcal{T}$ και γράφουμε $\mathcal{I} \models \mathcal{T}$.

Τέλος, μπορούμε να δημιουργήσουμε και αξιώματα ατόμων, που ονομάζονται *σχέσεις στιγμιοτύπου* (*instance relations*) ή αλλιώς *ισχυρισμοί* (*assertions*) ανάμεσα σε άτομα (ζεύγη ατόμων) και έννοιες (ρόλους) τα οποία συλλέγονται στα σώματα ισχυρισμών. Ένα DL-Lite$_{core}$ *σώμα ισχυρισμών* (*assertional box – ABox*), συμβολίζεται με $\mathcal{A}$ και αναπαριστά ένα σύνολο από ισχυρισμούς της μορφής $a : A$ ή $A(a)$ που ονομάζονται *ισχυρισμοί εννοιών* (*concept assertions*) και $(a,b) : P$ ή $P(a,b)$ που ονομάζονται *ισχυρισμοί ρόλων* (*role assertions*), όπου τα a, b είναι άτομα, η A μια ατομική έννοια και ο P ένας ατομικός ρόλος. Οι ισχυρισμοί αυτοί δηλώνουν ότι το άτομο a αποτελεί στιγμιότυπο της έννοιας A και ότι τα άτομα a, b αποτελούν στιγμιότυπα του ρόλου P. Μια ερμηνεία $\mathcal{I}$ ικανοποιεί τον ισχυρισμό $a : A$ εάν $a^{\mathcal{I}} \in A^{\mathcal{I}}$ και τον ισχυρισμό $(a,b) : P$ εάν $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in P^{\mathcal{I}}$. Μια ερμηνεία ικανοποιεί ένα DL-Lite$_{core}$ ABox $\mathcal{A}$ εάν ικανοποιεί κάθε αξίωμα στο $\mathcal{A}$. Στην περίπτωση αυτή λέμε ότι η $\mathcal{I}$ είναι μοντέλο

του $\mathcal{A}$ και γράφουμε $\mathcal{I} \models \mathcal{A}$.

Μια DL-Lite$_{core}$ *οντολογία* (*ontology*) $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ αποτελείται από ένα TBox και από ένα ABox. Μια ερμηνεία ικανοποιεί μια DL-Lite$_{core}$ οντολογία $\mathcal{O}$, και θα γράφουμε $\mathcal{I} \models \mathcal{O}$, εάν ικανοποιεί κάθε αξίωμα της $\mathcal{O}$, εάν δηλαδή $\mathcal{I} \models \mathcal{T}$ και $\mathcal{I} \models \mathcal{A}$. Στην περίπτωση αυτή λέμε ότι η $\mathcal{I}$ είναι μοντέλο της $\mathcal{O}$.

2.4.1 DL-Lite$_F$ και DL-Lite$_R$

Στη συνέχεια παρουσιάζουμε δύο Περιγραφικές Λογικές που επεκτείνουν την DL-Lite$_{core}$. Η πρώτη από αυτές είναι η DL-Lite$_F$ που επεκτείνει την DL-Lite$_{core}$ με την ικανότητα να δηλώνουμε *συναρτησιακούς ρόλους* (*functional roles*). Για ένα βασικό ρόλο R ο ισχυρισμός

$$(\mathsf{func}\ R)$$

δηλώνει ότι ο ρόλος R είναι συναρτησιακός. Μια ερμηνεία $\mathcal{I}$ αποτελεί μοντέλο του ισχυρισμού ($\mathsf{func}\ R$) εάν η δυαδική σχέση $R^{\mathcal{I}}$ είναι μια συνάρτηση, ήτοι εάν $(a, b_1) \in R^{\mathcal{I}}$ και $(a, b_2) \in R^{\mathcal{I}}$ συνεπάγονται ότι $b_1 = b_2$.

Η δεύτερη γλώσσα επεκτείνει την DL-Lite$_{core}$ με τον ορισμό *αξιωμάτων υπαγωγής ρόλων* (*role inclusion axioms*) είναι η DL-Lite$_R$. Τα αξιώματα αυτά είναι της μορφής

$$R \sqsubseteq E$$

όπου R ένας βασικός και E ένας γενικός ρόλος. Μια ερμηνεία $\mathcal{I}$ αποτελεί μοντέλο ενός αξιώματος υπαγωγής $R \sqsubseteq E$ εάν $R^{\mathcal{I}} \subseteq E^{\mathcal{I}}$. Η δυνατότητα ορισμού τέτοιων αξιωμάτων στην DL-Lite$_R$ επιτρέπει την προσομοίωση *προσδιορισμένων υπαρξιακών περιορισμών* (*qualified existential restriction*) στο δεξί μέλος των αξιωμάτων υπαγωγής εννοιών [33]. Έστω για παράδειγμα ότι έχουμε το αξίωμα υπαγωγής $B \sqsubseteq \exists R.C$. Το αξίωμα αυτό μπορεί να γραφεί ισοδύναμα ως το σύνολο των παρακάτω αξιωμάτων

$$B \sqsubseteq \exists R_{aux} \qquad \exists R_{aux}^{-} \sqsubseteq C \qquad R_{aux} \sqsubseteq R$$

όπου R_{aux} ένας βοηθητικός ρόλος που δεν εμφανίζεται μέσα στην οντολογία.

Όπως φαίνεται από τα παραπάνω η οικογένεια γλωσσών της DL-Lite είναι αρκετά απλή. Παρά την απλότητα της όμως μπορεί να συλλάβει τις βασικές ιδέες (όχι όμως όλες, προφανώς) τόσο των οντολογιών όσο και των φορμαλισμών μοντελοποίησης εννοιών που χρησιμοποιούνται στις βάσεις δεδομένων (ήτοι, το μοντέλο Οντοτήτων-Συσχετίσεων). Πιο συγκεκριμένα, οι ισχυρισμοί της DL-Lite μας δίνουν τη δυνατότητα να ορίσουμε τις σχέσεις:

- *isa*, να δηλώσουμε δηλαδή ότι μια έννοια A_1 υπάγεται από μια άλλη έννοια A_2, με τη χρήση του αξιώματος υπαγωγής $A_1 \sqsubseteq A_2$,
- *ξένων εννοιών*, να δηλώσουμε δηλαδή ότι δύο έννοιες A_1 και A_2 είναι ξένες μεταξύ τους, με τη χρήση του αξιώματος υπαγωγής $A_1 \sqsubseteq \neg A_2$,
- *τύπου ρόλου* (*role-typing*), να δηλώσουμε δηλαδή ότι το πρώτο (δεύτερο) συστατικό του ρόλου P είναι στιγμιότυπο της έννοιας A_1 (A_2), με τη χρήση του αξιώματος $\exists P \sqsubseteq A_1$ ($\exists P^{-} \sqsubseteq A_2$),

Πίνακας 2.2: Μετάφραση εννοιών και ρόλων σε φόρμουλες Λογικής Πρώτης Τάξης.

$$\begin{aligned}
\gamma_x(B) &= B(x) \\
\gamma_{x,y}(R) &= R(x,y) \\
\gamma_{x,y}(R^-) &= R(y,x) \\
\gamma_x(\exists R) &= \exists y.\gamma_{x,y}(R)
\end{aligned}$$

Πίνακας 2.3: Μετάφραση αξιωμάτων και οντολογιών σε προτάσεις Λογικής Πρώτης Τάξης.

$$\begin{aligned}
\gamma(B_1 \sqsubseteq B_2) &= \forall x.(\gamma_x(B_1) \rightarrow \gamma_x(B_2)) \\
\gamma(R_1 \sqsubseteq R_2) &= \forall x,y.(\gamma_{x,y}(R_1) \rightarrow \gamma_{x,y}(R_2)) \\
\gamma(\mathsf{func}\ R_1) &= \forall x,y_1,y_2.(\gamma_{x,y_1}(R_1) \wedge \gamma_{x,y_2}(R_1) \rightarrow y_1 = y_2) \\
\gamma(A(a)) &= A(a) \\
\gamma(P(a,b)) &= P(a,b) \\
\gamma(\mathcal{O}) &= \bigwedge_{I \in \mathcal{O}} \gamma(I)
\end{aligned}$$

- *υποχρεωτικής συμμετοχής* (*mandatory participation*), να δηλώσουμε δηλαδή ότι όλα τα στιγμιότυπα μιας έννοιας A συμμετέχουν στον ρόλο P ως το πρώτο (δεύτερο) όρισμα, με τη χρήση του αξιώματος $A \sqsubseteq \exists P$ ($A \sqsubseteq \exists P^-$),

- *υποχρεωτικής μη-συμμετοχής* (*mandatory non-participation*), χρησιμοποιώντας τα $A \sqsubseteq \neg\exists P$ και $A \sqsubseteq \neg\exists P^-$,

- *συναρτησιακού περιορισμού* (*functionality restrictions*) σε ρόλους, με χρήση των αναθέσεων (func P) και (func P^-).

Εκτός από τη χρήση μοντελοθεωρητικής σημασιολογίας, όπως δείξαμε παραπάνω, οι γλώσσες που ανήκουν στην οικογένεια γλωσσών της DL-Lite μπορούν να ερμηνευθούν και με τη χρήση της σημασιολογίας της Λογικής Πρώτης Τάξης, μέσω του μετασχηματισμού τους σε υπαρξιακούς κανόνες. Στους Πίνακες 2.2 και 2.3 παρουσιάζουμε τους μετασχηματισμούς αυτούς για τις γλώσσες DL-Lite$_F$ και DL-Lite$_R$, όπου $\gamma(\cdot)$ είναι η συνάρτηση μετάφρασης.

Σημειώνουμε τέλος ότι η DL-Lite$_F$ αποτελεί αυστηρό υποσύνολο της γλώσσας οντολογιών OWL[2], και πιο συγκεκριμένα της OWL Lite, ενώ η DL-Lite$_R$ αποτελεί το λογικό υπόβαθρο της γλώσσας OWL 2 QL, που αποτελεί ένα γνωστό βατό προφίλ της OWL 2 [104].

2.4.2 Υπηρεσίες Συλλογιστικής

Όπως αναφέρθηκε και στην εισαγωγή ένα σύστημα αναπαράστασης γνώσης παρέχει τη δυνατότητα όχι μόνο της αποθήκευσης γνώσης, αλλά και της χρήσης *υπηρεσιών συλλογιστικής* (*inference services*) με τις οποίες μπορούμε να εξάγουμε νέα γνώση

[2] `http://www.w3.org/TR/owl-features/`

και νέα συμπεράσματα από τη γνώση που έχουμε περιγράψει. Οι ΠΛ μας προσφέρουν γενικά τις παρακάτω υπηρεσίες:

Ικανοποιησιμότητα μιας οντολογίας: Μια οντολογία $\mathcal{O}$ είναι *ικανοποιήσιμη* (*satisfiable*) ανν υπάρχει μοντέλο $\mathcal{I}$ για την $\mathcal{O}$, δηλαδή αν και μόνο αν (ανν) $\mathcal{I} \models \mathcal{O}$. Αντίστοιχα ορίζεται και η έννοια της μη-ικανοποιησιμότητας (unsatisfiability).

Ικανοποιησιμότητα μιας έννοιας: Μια έννοια C είναι ικανοποιήσιμη μ.β.τ. $\mathcal{O}$ ανν υπάρχει ερμηνεία $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ και ένα στοιχείο $a \in \Delta^{\mathcal{I}}$ τ.ω. $\mathcal{I} \models \mathcal{O}$ και $a \in C^{\mathcal{I}}$.

Υπαγωγή μιας έννοιας σε μια άλλη: Μια έννοια D *υπάγει* (*subsumes*) μια έννοια C εάν για κάθε μοντέλο $\mathcal{I}$ τ.ω. $\mathcal{I} \models \mathcal{O}$ έχουμε $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$.

Συνέπεια ενός σώματος ισχυρισμών: Ένα σώμα ισχυρισμών (ABox) $\mathcal{A}$ είναι συνεπές εάν υπάρχει κάποιο μοντέλο $\mathcal{I}$ τ.ω. $\mathcal{I} \models \mathcal{A}$.

Τα προβλήματα συλλογιστικής αυτά τίθενται υπό μορφή ερώτησης στη μηχανή συλλογιστικής και μπορούν να αναχθούν σε μια ερώτηση ανάκτησης στιγμιοτύπων για μια έννοια ή ρόλο. Παρόλα αυτά είναι ιδιαίτερα συχνό το φαινόμενο που οι χρήστες επιθυμούν να υποβάλλουν πιο σύνθετες ερωτήσεις οι οποίες να μη μπορούν να αναχθούν σε μια τέτοια ερώτηση ανάκτησης στιγμιοτύπων. Η εκφραστικότητα για την υποβολή τέτοιων σύνθετων ερωτήσεων παρέχεται από τα *Συζευκτικά Ερωτήματα (ΣΕ)*.

Έχοντας ορίσει τα ΣΕ μπορούμε τώρα να ορίσουμε τα ΣΕ πάνω σε μια οντολογία ΠΛ. Ένα ΣΕ πάνω σε μια οντολογία $\mathcal{O}$ είναι ένα ΣΕ τα άτομα του οποίου είναι της μορφής $A(z)$ ή $P(z_1, z_2)$, με A, P ατομικές έννοιες και ατομικούς ρόλους αντίστοιχα που εμφανίζονται στην $\mathcal{O}$, και z, z_1, z_2 είτε σταθερές που εμφανίζονται στην $\mathcal{O}$ είτε μεταβλητές. Με τον ίδιο τρόπο μπορούμε να ορίσουμε και ένα ΣΣΕ πάνω σε μια οντολογία $\mathcal{O}$.

Μια *απάντηση* (*certain answer*) σε ένα ΣΕ $\mathcal{Q}$ σε σχέση με μια οντολογία $\mathcal{O}$ είναι μια πλειάδα $\vec{c} = (c_1, \ldots, c_n)$ από άτομα τ.ω. η $\mathcal{O}$ να *επάγει* (*entails*) τη φόρμουλα Λογικής Πρώτης Τάξης που κατασκευάζεται από τη σύζευξη όλων των ατόμων του σώματος στο $\mathcal{Q}$, αντικαθιστώντας κάθε διακεκριμένη μεταβλητή x_j με c_j και κάθε μη-διακεκριμένη με μια υπαρξιακή μεταβλητή. Ορίζουμε με $\mathsf{cert}(\mathcal{Q}, \mathcal{O})$ το σύνολο όλων των απαντήσεων του $\mathcal{Q}$ μ.β.τ. $\mathcal{O}$.

Με βάση τα παραπάνω μπορούμε να ορίσουμε ακόμα μια υπηρεσία συλλογιστικής, την *απάντηση ερωτημάτων* (*query answering*) σύμφωνα με την οποία για μια οντολογία $\mathcal{O}$ και ένα ερώτημα $\mathcal{Q}$ στην $\mathcal{O}$ πρέπει να υπολογιστεί το σύνολο $\mathsf{cert}(\mathcal{Q}, \mathcal{O})$.

2.5 Απάντηση Ερωτημάτων μέσω Επανεγγραφής

Η *απάντηση ερωτημάτων* (*query answering*) γίνεται μέσω μιας διαδικασίας που ονομάζεται *επανεγγραφή ερωτημάτων* (*query rewriting*) [31, 117] και στηρίζεται στην

ιδιότητα της *επανεγγραψιμότητας* (*rewritability*). Διαισθητικά, η επανεγγραψιμότητα εκφράζει την ιδιότητα σύμφωνα με την οποία η απάντηση ερωτημάτων με βάση μια οντολογία (TBox και ABox) μπορεί να αναχθεί στον υπολογισμό ενός ερωτήματος που ονομάζεται επανεγγραφή σε ένα ABox $\mathcal{A}$ το οποίο θεωρούμε ως μια (επαγωγική) βάση δεδομένων.

Έτσι λοιπόν, δεδομένου ενός TBox $\mathcal{T}$ και ενός ερωτήματος $\mathcal{Q}$, με την τεχνική της επανεγγραφής ερωτημάτων υπολογίζεται ένα νέο ερώτημα, που ονομάζεται *επανεγγραφή* (*rewriting*). Ανάλογα με την εκφραστικότητα της ΠΛ η επανεγγραφή χρειάζεται να εκφραστεί σε διαφορετικούς φορμαλισμούς. Εάν το $\mathcal{T}$ είναι εκφραστικότητας $\mathcal{ELHIO}$, τότε η επανεγραφή ενός ερωτήματος $\mathcal{Q}$ μ.β.τ. $\mathcal{T}$ είναι στη χειρότερη περίπτωση μια επανεγγραφή Datalog, εάν το $\mathcal{T}$ είναι εκφραστικότητας γραμμικής Datalog$^{\pm}$ [26, 66], τότε η επανεγγραφή ενός ερωτήματος $\mathcal{Q}$ μ.β.τ. $\mathcal{T}$ είναι στη χειρότερη περίπτωση μια επανεγγραφή μη-αναδρομικής Datalog, εάν το $\mathcal{T}$ είναι εκφραστικότητας DL-Lite^{+} [6], τότε η επανεγγραφή αποτελείται στη χειρότερη περίπτωση από μια επανεγγραφή ΣΣΕ και από μια επανεγγραφή γραμμικής Datalog, ενώ τέλος εάν το $\mathcal{T}$ είναι εκφραστικότητας DL-Lite$_R$, τότε η επανεγγραφή ενός ερωτήματος $\mathcal{Q}$ μ.β.τ. $\mathcal{T}$ είναι στη χειρότερη περίπτωση μια επανεγγραφή ΣΣΕ. Το πρόβλημα της αποτίμησης ενός προγράμματος Datalog εμφανίζει πολυπλοκότητα PTIME ως προς τα δεδομένα, ενώ αυτό της απάντησης ενός ΣΣΕ ανήκει στην κλάση πολυπλοκότητας AC$_0$ [112, 166, 165, 54], που περιέχεται στην LOGSPACE, ως προς τα δεδομένα. Παρατηρούμε δηλαδή, ότι η απάντηση ερωτημάτων για βατές ΠΛ είναι χαμηλής πολυπλοκότητας. Στη συνέχεια ορίζουμε την επανεγγραφή Datalog που καλύπτει όλες τις προηγούμενες περιπτώσεις.

Ορισμός 2.5.1. Μια *επανεγγραφή Datalog* (*Datalog rewriting*) για ένα ΣΕ $\mathcal{Q}$ μ.β.τ. TBox $\mathcal{T}$ είναι ένα πρόγραμμα Datalog $\mathcal{R}$ το οποίο μπορεί να χωριστεί σε δύο ανεξάρτητα σύνολα $\mathcal{R}_D \uplus \mathcal{R}_Q$, με $\mathcal{R}_D$ ένα σύνολο από κανόνες Datalog που δεν περιέχουν το κατηγόρημα $\mathcal{Q}_A$ και $\mathcal{R}_Q$ ένα ΣΣΕ με κατηγόρημα ερωτήματος $\mathcal{Q}_A$ του οποίου τα άτομα του σώματος αναφέρουν μόνο κατηγορήματα που περιέχονται στο $\mathcal{T}$, και που για κάθε ABox $\mathcal{A}$ με τη χρήση κατηγορημάτων από το $\mathcal{T}$ έχουμε

$$\mathsf{cert}(\mathcal{Q}, \mathcal{T} \cup \mathcal{A}) = \mathsf{cert}(\mathcal{R}_Q, \mathcal{R}_D \cup \mathcal{A}). \tag{2.3}$$

Η επανεγγραφή $\mathcal{R}$ είναι μια *επανεγγραφή ΣΣΕ* (*UCQ rewriting*) εάν $\mathcal{R}_D = \emptyset$. Στην περίπτωση αυτή αναφερόμαστε στην επανεγγραφή χρησιμοποιώντας μόνο το σύνολο $\mathcal{R}_Q$. △

Μια επανεγγραφή μπορεί να υπολογιστεί εφαρμόζοντας μετασχηματισμούς που διατηρούν την ισοδυναμία (equivalence preserving transformations) για τα δεδομένα εισόδου $\mathcal{T}$ και $\mathcal{Q}$. Διάφορες τεχνικές και αλγόριθμοι έχουν προταθεί μέχρι στιγμής στην βιβλιογραφία για διάφορες γλώσσες οντολογιών όπως είναι η DL-Lite, η $\mathcal{ELHI}$, τα γραμμικά TGDs και άλλες [31, 117, 132, 40, 64, 87, 108, 25] και πολλοί από αυτούς διαφέρουν σημαντικά μεταξύ τους. Στη συνέχεια και για να μπορέσουμε να αποστασιοποιηθούμε από τις ιδιαιτερότητες του κάθε αλγορίθμου επανεγγραφής, παρουσιάζουμε έναν γενικό ορισμό ο οποίος συλλαμβάνει την έννοια των περισσότερων αλγορίθμων επανεγγραφής που εμφανίζονται στη βιβλιογραφία χρησιμοποιώντας κανόνες συμπερασμού.

Ορισμός 2.5.2. Ένας *αλγόριθμος επανεγγραφής* (*rewriting algorithm*) rew για μια ΠΛ $\mathcal{L}$ είναι ένας αλγόριθμος ο οποίος για κάθε $\mathcal{L}$-οντολογία $\mathcal{T}$ και ΣΕ $\mathcal{Q}$:

1. Μετασχηματίζει το TBox $\mathcal{T}$ σε ένα σύνολο από προτάσεις N_1 στο οποίο προσθέτει και το ερώτημα $\mathcal{Q}$.

2. Χρησιμοποιεί ένα σύστημα συμπερασμού Γ για να υπολογίσει μια ακολουθία $\langle N_1, \sigma_1, \Upsilon_1\rangle \triangleright \cdots \triangleright \langle N_i, \sigma_i, \Upsilon_i\rangle$ πλειάδων κάθε μια από τις οποίες αποτελείται από ένα σύνολο από προτάσεις N_i, μια αντικατάσταση σ_i και μια λίστα από ζεύγη Υ_i για τα οποία για $i = 1$ έχουμε $\sigma_1 = \emptyset$, $\Upsilon_1 = \{\langle\emptyset, \emptyset\rangle\}$. Επιπλέον, για κάθε $i \in \{1, \ldots, m-1\}$, το N_{i+1} επεκτείνει το N_i με ένα συμπέρασμα ενός συμπερασμού στο $\Gamma(N_i)$ και σ_{i+1}, Υ_{i+1} είναι οι υποσημειώσεις του συγκεκριμένου συμπερασμού.

3. Επιστρέφει μια πλειάδα $\mathsf{rew}(\mathcal{Q}, \mathcal{O}) = \langle \mathcal{R}, \rightsquigarrow_{\mathcal{R}}\rangle$, όπου $\mathcal{R} = \mathcal{R}_D \uplus \mathcal{R}_Q$ είναι υποσύνολο του N_m τ.ω. να είναι επανεγγραφή για το $\mathcal{Q}$ μ.β.τ. $\mathcal{T}$, και $\rightsquigarrow_{\mathcal{R}}$ είναι η σχέση που ορίζεται ως εξής: για $x \in \mathsf{var}(\mathcal{Q})$, $\mathcal{Q}' \in \mathcal{R}_Q$, $y \in \mathsf{var}(\mathcal{Q}')$, έχουμε ότι $x \rightsquigarrow_{\mathcal{R}} y$ ανν $x(\sigma_1 \circ \ldots \circ \sigma_j) = y$, όπου j είναι ο μικρότερος ακέραιος για τον οποίο ισχύει $\mathcal{Q}' \in N_j$.

Ένας αλγόριθμος επανεγγραφής ονομάζεται *Q-προσανατολισμένος* (*Q-oriented*) εάν όλα τα $\mathcal{Q}_i \in \mathcal{R}_Q$ που είναι διαφορετικά από το $\mathcal{Q}$ παράγονται από κάποιο $\mathcal{Q}_j \in \mathcal{R}_Q$ από ένα συμπερασμό της μορφής $\langle \mathcal{Q}_j, C_1, \ldots, C_n, \mathcal{Q}_i\rangle$. △

Οι περισσότεροι αλγόριθμοι επανεγγραφής είτε στηρίζονται στον κανόνα της ανάλυσης ή μπορούν να αναχθούν εύκολα στο πλαίσιο της ανάλυσης, και συνεπώς ο παραπάνω ορισμός ισχύει για αυτούς. Για τη συνέχεια της διατριβής θα θεωρήσουμε επίσης ότι οι επανεγγραφές υπολογίζονται από αλγόριθμους που είναι Q-προσανατολισμένοι. Πάρα πολλοί διάσημοι αλγόριθμοι και συστήματα επανεγγραφής, όπως το PerfectRef, το Nyaya, και το Rapid, στηρίζονται σε αλγόριθμους που είναι Q-προσανατολισμένοι. Παρόλα αυτά, άλλα πολύ γνωστά συστήματα, όπως το Requiem, δεν είναι Q-προσανατολισμένα. Για παράδειγμα στο Requiem ένα ερώτημα $\mathcal{Q}_i \in \mathcal{R}_Q$ μπορεί να υπολογιστεί από ένα ερώτημα $\mathcal{Q}_j \in \mathcal{R}_Q$ χρησιμοποιώντας περισσότερους του ενός συμπερασμούς που περιλαμβάνουν ενδιάμεσες προτάσεις που περιέχουν το κατηγόρημα του ερωτήματος αλλά περιέχουν επίσης και συναρτησιακούς όρους. Για παράδειγμα, για $\mathcal{T} = \{A(x) \to \exists y.R(x,y) \wedge C(y), R(x,y) \to S(x,y)\}$ και $\mathcal{Q} = S(x,y) \wedge C(y) \to \mathcal{Q}_A(x)$ το Requiem πράττει τα ακόλουθα:

- Αρχικά, μετασχηματίζει το TBox $\mathcal{T}$ στο παρακάτω σύνολο από προτάσεις:

$$A(x) \;\to\; R(x, f(x)) \tag{2.4}$$
$$A(x) \;\to\; C(f(x)) \tag{2.5}$$
$$R(x,y) \;\to\; S(x,y) \tag{2.6}$$

  όπου f είναι μια νέα συνάρτηση skolem [138].

- Στη συνέχεια, εφαρμόζει τον κανόνα της ανάλυσης με ελεύθερη επιλογή (resolution with free selection) και μια πιθανή κατάλληξη μπορεί να είναι η παρακάτω: από το $\mathcal{Q}$ και την εξίσωση (2.5) να προκύψει το $\mathcal{Q}_f^1 = S(x, f(x)) \wedge A(x) \rightarrow \mathcal{Q}_A(x)$, από το $\mathcal{Q}_f^1$ και την εξίσωση (2.6) να προκύψει το $\mathcal{Q}_f^2 = R(x, f(x)) \wedge A(x) \rightarrow \mathcal{Q}_A(x)$, και τέλος, από το $\mathcal{Q}_f^2$ και την εξίσωση (2.4) να προκύψει το $\mathcal{Q}' = A(x) \rightarrow \mathcal{Q}_A(x)$. Η επανεγγραφή του $\mathcal{Q}$ μ.β.τ. $\mathcal{T}$ είναι το σύνολο από τις προτάσεις που δεν περιέχουν συναρτησιακούς όρους, δηλαδή, $\mathcal{R} = \{\mathcal{Q}, \mathcal{Q}'\}$.

Είναι προφανές ότι ο αλγόριθμος Requiem δεν είναι Q-προσανατολισμένος καθώς το $\mathcal{Q}'$ προκύπτει από το ερώτημα $\mathcal{Q}$ μέσα από περισσότερα του ενός βήματα που περιλαμβάνουν ενδιάμεσες προτάσεις που περιέχουν συναρτησιακούς όρους και οι οποίες δεν ανήκουν στην τελική επανεγγραφή.

Παρόλα αυτά τα αποτελέσματα που παρουσιάζουμε στη συνέχεια της διατριβής μας μπορούν να εφαρμοστούν και σε τέτοια συστήματα χρησιμοποιώντας την παρακάτω προσέγγιση: Αντί να δεχόμαστε σαν είσοδο μια επανεγγραφή $\mathcal{R}$, οι αλγόριθμοι που θα παρουσιάσουμε μπορούν να δέχονται σαν είσοδο το όριο N_m της ακολουθίας παραγωγής που προκύπτει από τον αλγόριθμο επανεγγραφής. Για παράδειγμα, στην προηγούμενη περίπτωση οι αλγόριθμοι μας θα δεχθούν σαν είσοδο το σύνολο $N_3 = \{\mathcal{Q}, \mathcal{Q}_f^1, \mathcal{Q}_f^2, \mathcal{Q}'\}$. Αντικαθιστώντας μια επανεγγραφή με το όριο N_m σε όλους τους ορισμούς και όλους τους αλγορίθμους τα αποτελέσματα εφαρμόζονται χωρίς περαιτέρω τροποποιήσεις.

2.5.1 Ο Αλγόριθμος Επανεγγραφής PerfectRef

Στη συνέχεια παρουσιάζουμε τον αλγόριθμο επανεγγραφής PerfectRef που προτάθηκε από τους Calvanese et al. [31] καθώς μέρος των αλγορίθμων που περιγράφουμε στη συνέχεια της διατριβής μας στηρίζεται σε αυτόν. Ο αλγόριθμος αυτός, που φαίνεται στον Αλγόριθμο 2, εφαρμόζει εξαντλητικά δύο βήματα: ένα *βήμα αναδιατύπωσης* (*reformulation step*) (γραμμές 8 – 16) και ένα *βήμα μείωσης* (*reduction step*) (γραμμές 17 – 22). Κάθε βήμα παίρνει ως είσοδο ένα ΣΕ και πιθανώς κάποιο αξίωμα I από το $\mathcal{T}$ και παράγει ένα νέο ΣΕ. Η διαδικασία τερματίζει όταν δεν παράγεται κανένα νέο ερώτημα.

Πιο συγκεκριμένα, στο βήμα αναδιατύπωσης, για ένα άτομο α ενός ερωτήματος $\mathcal{Q}$ και ένα αξίωμα $I \in \mathcal{T}$, η συνάρτηση $\mathsf{gr}(\alpha, I)$ επιστρέφει ένα νέο άτομο. Πιο τυπικά:

Ορισμός 2.5.3. Έστω I αξίωμα υπαγωγής και α άτομο του ερωτήματος $\mathcal{Q}$. Τότε η συνάρτηση $\mathsf{gr}(\alpha, I)$ ορίζεται ως εξής:

- Εάν $\alpha = A(x)$ και
  1. $I = B(x) \rightarrow A(x)$, τότε $\mathsf{gr}(\alpha, I) = B(x)$,
  2. $I = \exists y.P(x, y) \rightarrow A(x)$, τότε $\mathsf{gr}(\alpha, I) = P(x, y)$ για μια νέα μεταβλητή y στο $\mathcal{Q}$,
  3. $I = \exists y.P(y, x) \rightarrow A(x)$, τότε $\mathsf{gr}(\alpha, I) = P(y, x)$ για μια νέα μεταβλητή y στο $\mathcal{Q}$.

Αλγόριθμος 2 PerfectRef($\mathcal{Q}, \mathcal{T}$)

Είσοδος: Ένα ΣΕ $\mathcal{Q}$ και ένα DL-Lite$_R$-TBox $\mathcal{T}$

1: Αρχικοποίησε μια επανεγγραφή ΣΣΕ $\mathcal{R}'_Q := \{\mathcal{Q}\}$
2: **repeat**
3: $\quad\mathcal{R}_Q := \mathcal{R}'_Q$
4: $\quad$**for all** $\mathcal{Q}_1 \in \mathcal{R}_Q$ **do**
5: $\quad\quad$**for all** $\alpha \in \mathcal{Q}_1$ **do**
6: $\quad\quad\quad$**for all** $I \in \mathcal{T}$ **do**
7: $\quad\quad\quad\quad$**if** I είναι εφαρμόσιμο στο α **then**
8: $\quad\quad\quad\quad\quad$Πρόσθεσε το $\mathcal{Q}_2 := \mathcal{Q}_1[\alpha/\mathsf{gr}(\alpha, I)]$ στο $\mathcal{R}'_Q$
9: $\quad\quad\quad\quad$**end if**
10: $\quad\quad\quad$**end for**
11: $\quad\quad$**end for**
12: $\quad\quad$**for all** α_1, α_2 στο $\mathcal{Q}_1$ **do**
13: $\quad\quad\quad$**if** υπάρχει mgu σ για τα α_1 και α_2 **then**
14: $\quad\quad\quad\quad$Πρόσθεσε το $\mathcal{Q}_1\sigma$ στο $\mathcal{R}'_Q$
15: $\quad\quad\quad$**end if**
16: $\quad\quad$**end for**
17: $\quad$**end for**
18: **until** $\mathcal{R}_Q = \mathcal{R}'_Q$
19: **return** $\mathcal{R}_Q$

- Εάν $\alpha = P(x, z)$, με z μη-δεσμευμένη στο $\mathcal{Q}$ και
  1. $I = A(x) \rightarrow \exists y.P(x, y)$, τότε $\mathsf{gr}(\alpha, I) = A(x)$,
  2. $I = \exists y.S(x, y) \rightarrow \exists z.P(x, z)$, τότε $\mathsf{gr}(\alpha, I) = S(x, y)$ για μια νέα μεταβλητή y στο $\mathcal{Q}$,
  3. $I = \exists y.S(y, x) \rightarrow \exists z.P(x, z)$, τότε $\mathsf{gr}(\alpha, I) = S(y, x)$, για μια νέα μεταβλητή y στο $\mathcal{Q}$.
- Εάν $\alpha = P(z, x)$, με z μη-δεσμευμένη στο $\mathcal{Q}$, και
  1. $I = A(x) \rightarrow \exists z.P(z, x)$, τότε $\mathsf{gr}(\alpha, I) = A(x)$,
  2. $I = \exists y.S(x, y) \rightarrow \exists z.P(z, x)$, τότε $\mathsf{gr}(\alpha, I) = S(x, y)$, για μια νέα μεταβλητή y στο $\mathcal{Q}$,
  3. $I = \exists y.S(y, x) \rightarrow \exists z.P(z, x)$, τότε $\mathsf{gr}(\alpha, I) = S(y, x)$, για μια νέα μεταβλητή y στο $\mathcal{Q}$.
- Εάν $\alpha = P(x, y)$ και
  1. $I = S(x, y) \rightarrow P(x, y)$, τότε $\mathsf{gr}(\alpha, I) = S(x, y)$;
  2. $I = S(y, x) \rightarrow P(x, y)$, τότε $\mathsf{gr}(\alpha, I) = S(y, x)$.

$\triangle$

Εάν για κάποιο αξίωμα I και κάποιο άτομο α η συνάρτηση $\mathsf{gr}(\alpha, I)$ επιστρέφει κάποια τιμή, όπως φαίνεται και από τον Ορισμό 2.5.3, τότε λέμε ότι το I είναι *εφαρμόσιμο* (*applicable*) στο α, και η εφαρμογή αυτή σε κάποιο ΣΕ $\mathcal{Q}$ παράγει ένα νέο ΣΕ της μορφής $\mathcal{Q}[\alpha/\mathsf{gr}(\alpha, I)]$, δηλαδή ένα νέο ερώτημα που περιέχει το άτομο $\mathsf{gr}(\alpha, I)$ αντί του ατόμου α. Για παράδειγμα, για το ΣΕ $\mathcal{Q}_3 = \mathsf{διδάσκει}(x, y) \wedge \mathsf{διδάσκει}(y, z) \rightarrow \mathcal{Q}_A(x)$ και το αξίωμα $I_1 = \mathsf{ΜεταπτυχιακόςΦοιτητής}(y) \rightarrow \exists z.\mathsf{διδάσκει}(y, z)$ το βήμα αναδιατύπωσης αντικαθιστά το άτομο $\mathsf{διδάσκει}(y, z)$ με το $\mathsf{ΜεταπτυχιακόςΦοιτητής}(y)$ παράγοντας το νέο ερώτημα $\mathcal{Q}_4 = \mathsf{διδάσκει}(x, y) \wedge \mathsf{ΜεταπτυχιακόςΦοιτητής}(y) \rightarrow \mathcal{Q}_A(x)$. Παρατηρούμε ότι το αξίωμα I_1 δεν είναι εφαρμόσιμο στο άτομο $\mathsf{διδάσκει}(x, y)$ εφόσον η μεταβλητή y είναι δεσμευμένη στο ερώτημα $\mathcal{Q}_3$. Στη συνέχεια, για το αξίωμα της μορφής $I_2 = \exists w.\mathsf{διδάσκει}(w, y) \rightarrow \mathsf{ΜεταπτυχιακόςΦοιτητής}(y)$ το βήμα αναδιατύπωσης αντικαθιστά το άτομο $\mathsf{ΜεταπτυχιακόςΦοιτητής}(y)$ στο $\mathcal{Q}_4$ παράγοντας το Συζευκτικό Ερώτημα $\mathcal{Q}_5 = \mathsf{διδάσκει}(x, y) \wedge \mathsf{διδάσκει}(w, y) \rightarrow \mathcal{Q}_A(x)$, με w μια νέα μεταβλητή που δεν εμφανίζεται στο $\mathcal{Q}_4$.

Στο βήμα μείωσης, παράγεται ένα νέο ΣΕ από την εφαρμογή του πιο γενικού ενοποιητή (most general unifier – mgu) ανάμεσα σε δύο από τα άτομα κάποιου ΣΕ $\mathcal{Q}$. Το βήμα αυτο χρησιμοποιείται επειδή κάποια αξιώματα μπορούν να εφαρμοστούν μόνο στη μείωση ενός ερωτήματος, αφού με αυτό μια μεταβλητή που ήταν δεσμευμένη μπορεί να γίνει μη-δεσμευμένη. Για παράδειγμα, έστω το ερώτημα $\mathcal{Q}_5$. Παρατηρούμε ότι το αξίωμα I_1 δεν μπορεί να εφαρμοστεί στο $\mathcal{Q}_5$ αφού η μεταβλητή y (του $\mathcal{Q}_5$) είναι δεσμευμένη. Εφαρμόζοντας όμως το βήμα της μείωσης παράγεται το νέο ερώτημα $\mathcal{Q}_6 = \mathsf{διδάσκει}(x, y) \rightarrow \mathcal{Q}_A(x)$, εφόσον το άτομο $\mathsf{διδάσκει}(w, y)$ ενοποιείται με το $\mathsf{διδάσκει}(x, y)$, με τον ενοποιητή $w \mapsto x$ και στη συνέχεια το αξίωμα I_1 είναι εφαρμόσιμο παράγοντας το ερώτημα $\mathcal{Q}_7 = \mathsf{ΜεταπτυχιακόςΦοιτητής}(x) \rightarrow \mathcal{Q}_A(x)$. Παρόλα αυτά υπάρχουν περιπτώσεις που ενώ εφαρμόζεται το βήμα της μείωσης καμία δεσμευμένη μεταβλητή δεν γίνεται μη-δεσμευμένη. Όπως για παράδειγμα στο ΣΕ $\mathcal{Q}_8 = \mathsf{διδάσκει}(x, y) \wedge \mathsf{διδάσκει}(y, z) \rightarrow \mathcal{Q}_A(x)$ όπου το βήμα μείωσης παράγει το ερώτημα $\mathcal{Q}_9 = \mathsf{διδάσκει}(x, x) \rightarrow \mathcal{Q}_A(x)$ με τον πιο γενικό εννοποιητή $\{z \mapsto y, y \mapsto x\}$ στο οποίο δεν μπορεί να εφαρμοστεί το αξίωμα I_1. Αυτό έχει ως αποτέλεσμα την αύξηση του μεγέθους της επανεγγραφής ΣΣΕ με ερωτήματα τα οποία δεν πρόκειται να παράγουν νέα και ενδεχομένως είναι περιττά, όπως το ερώτημα $\mathcal{Q}_9$ που υπάγεται από το $\mathcal{Q}_8$.

Από τα παραπάνω, βλέπουμε ότι για το ερώτημα $\mathcal{Q}_3$ και το $\mathcal{T} = \{I_1, I_2\}$ ο αλγόριθμος PerfectRef παράγει την επανεγγραφή ΣΣΕ $\mathcal{R}_Q = \{\mathcal{Q}_3, \mathcal{Q}_4, \mathcal{Q}_5, \mathcal{Q}_6, \mathcal{Q}_7\}$. Ο αλγόριθμος αυτός είναι Q-προσανατολισμένος (Q-oriented) καθώς τα βήματα αναδιατύπωσης και μείωσης μπορούν να προκύψουν από βήματα συμπερασμού. Πιο συγκεκριμένα, το βήμα αναδιατύπωσης ουσιαστικά περιγράφει τη διαδικασία που ακολουθείται στον κανόνα συμπερασμού ανάλυσης αφού το άτομο α που αντικαθίσταται ουσιαστικά είναι το άτομο που ανήκει στο $\mathcal{A}_i$. Όσον αφορά στο βήμα μείωσης αυτό παρουσιάζει ομοιότητες με το βήμα παραγοντοποίησης στον οποίο άτομα από την κύρια υπόθεση ενοποιούνται με χρήση του πιο γενικού ενοποιητή που είναι ίσος με την αντικατάσταση σ.

2.6 Κατευθυνόμενοι Γράφοι

Ένας (κατευθυνόμενος) γράφος είναι μια πλειάδα $G = \langle V, E \rangle$, όπου V είναι ένα μη-κενό σύνολο από κόμβους και για το λόγο αυτό λέγεται *σύνολο κόμβων* και $E \subseteq V \times V$ ένα ενδεχομένως κενό σύνολο από δυαδικές σχέσεις, τις *ακμές* που ορίζονται πάνω στο V και λέγεται *σύνολο ακμών*. Στη συνέχεια της διατριβής μας κάνουμε συχνά κατάχρηση του συμβολισμού και χρησιμοποιούμε τον γράφο G για να αναφερθούμε στο σύνολο σχέσεων του γράφου. Στην περίπτωση αυτή το V είναι ακριβώς το σύνολο των κόμβων που συμμετέχουν στην E. Συνεπώς, προσθέτοντας ένα ζευγάρι $\langle a, b \rangle$ στον G προκύπτει ο νέος γράφος $G = \langle V \cup \{a, b\}, E \cup \{\langle a, b \rangle\} \rangle$.

Το *καρτεσιανό γινόμενο* (*cartesian product*) δύο γράφων G_1 και G_2 που απεικονίζεται με $G_1 \times G_2$ ορίζεται ως το καρτεσιανό γινόμενο του συνόλου των κόμβων τους. Δηλαδή, ορίζεται ως ο γράφος

$$G_1 \times G_2 = V_{G_1} \times V_{G_2} = \{(x, y) \mid x \in V_{G_1} \text{ και } y \in V_{G_2}\}$$

στον οποίο τα στοιχεία (x, y) και (x', y') σχηματίζουν ακμή, δηλαδή $\langle (x, y), (x', y') \rangle \in E_{G_1 \times G_2}$ εάν και μόνο εάν:

- $x = x'$ και $\langle y, y' \rangle \in E_{G_2}$, ή
- $y = y'$ και $\langle x, x' \rangle \in E_{G_1}$.

Για τους κόμβους $a, b \in V$, λέμε ότι ο b είναι *προσβάσιμος* (*reachable*) από τον a στον G, και γράφουμε $a \rightsquigarrow_G b$, εάν υπάρχουν $c_0, \ldots, c_n$ με $n \geq 0$ όπου $c_0 = a$, $c_n = b$ και $\langle c_i, c_{i+1} \rangle \in E$ για κάθε $0 \leq i < n$.[3] Ένα στοιχείο $c \in V$ καλείται *κόμβος κορυφής* (*top vertice*) στον G εάν για κάθε $c' \in V$ έχουμε $c \rightsquigarrow_G c'$. Τέλος, εάν ισχύει $\langle a, b \rangle \in G$ λέμε ότι ο κόμβος b είναι παιδί του κόμβου a.

[3]Σημειώνουμε ότι σύμφωνα με αυτό τον ορισμό κάθε κόμβος $c \in V$ είναι προσβάσιμος από τον εαυτό του.

Μέρος ΙΙ

Επανεγγραφή Τροποποιημένων Ερωτημάτων

Κεφάλαιο 3

Επανεγγραφή Τροποποιημένων Ερωτημάτων

Στο κεφάλαιο αυτό μελετάμε το πρόβλημα του υπολογισμού μιας επανεγγραφής για ερωτήματα τα οποία έχουν τροποποιηθεί. Για ένα συζευκτικό ερώτημα, ένα TBox και μια επανεγγραφή που έχει υπολογιστεί για αυτά, δείχνουμε πώς μπορούμε να υπολογίσουμε μια επανεγγραφή για ένα νέο ερώτημα το οποίο προκύπτει με τροποποίηση του αρχικού χρησιμοποιώντας την ήδη υπολογισμένη επανεγγραφή και αποφεύγοντας να υπολογίσουμε μια νέα εξαρχής. Πιο συγκεκριμένα, μελετάμε ερωτήματα τα οποία έχουν τροποποιηθεί με προσθήκη ή αφαίρεση διακεκριμένων μεταβλητών ή ατόμων. Για καλύτερη κατανόηση στην αρχή κάθε ενότητας δίνουμε μια σειρά από παραδείγματα που περιγράφουν διαισθητικά τους αλγορίθμους μας. Στη συνέχεια, τους παρουσιάζουμε αναλυτικά και αποδεικνύουμε την ορθότητα τους. Τέλος, παρουσιάζουμε μια σειρά από βελτιστοποιήσεις που έχουν ως στόχο την βελτίωση της απόδοσής τους.

3.1 Μείωση Ερωτημάτων με Διακεκριμένες Μεταβλητές

Στην ενότητα αυτή μελετάμε το πρόβλημα του υπολογισμού μιας επανεγγραφής για ένα ερώτημα του οποίου οι διακεκριμένες μεταβλητές έχουν μειωθεί, εκμεταλλευόμενοι όσο το δυνατόν περισσότερη πληροφορία από την ήδη υπάρχουσα επανεγγραφή για το αρχικό ερώτημα. Το παρακάτω παράδειγμα περιγράφει τις βασικές ιδέες στις οποίες στηρίζεται ο αλγόριθμος που θα παρουσιάσουμε στη συνέχεια.

Παράδειγμα 3.1.1. Έστω ένα TBox $\mathcal{T}$ που περιγράφει μαθήματα και καθηγητές:

$$\begin{aligned} \textsf{διδάσκειΜεταπτυχιακό}(x,y) &\rightarrow \textsf{διδάσκει}(x,y) \\ \textsf{Καθηγητής}(x) &\rightarrow \exists y.\textsf{διδάσκειΜεταπτυχιακό}(x,y). \end{aligned}$$

Έστω επίσης το ερώτημα

$$\mathcal{Q}_1 = \textsf{διδάσκει}(x_1,y_1) \rightarrow \mathcal{Q}_A(x_1,y_1)$$

το οποίο ανακτά όλους όσους διδάσκουν κάποιο μάθημα μαζί με το μάθημα αυτό. Το σύνολο $\mathcal{R} = \{\mathcal{Q}_1, \mathcal{Q}_2\}$, όπου $\mathcal{Q}_2 = \textsf{διδάσκειΜεταπτυχιακο}(x_2,y_2) \rightarrow \mathcal{Q}_A(x_2,y_2)$

αποτελεί επανεγγραφή του $\mathcal{Q}_1$ μ.β.τ. $\mathcal{T}$. Πιο συγκεκριμένα, μετασχηματίζοντας τα αξιώματα του $\mathcal{T}$ στις παρακάτω προτάσεις

$$\textsf{διδάσκειΜεταπτυχιακό}(x, y) \rightarrow \textsf{διδάσκει}(x, y) \tag{3.1}$$

$$\textsf{Καθηγητής}(x) \rightarrow \textsf{διδάσκειΜεταπτυχιακό}(x, f(x)) \tag{3.2}$$

έχουμε ότι το $\mathcal{Q}_2$ είναι το συμπέρασμα ενός συμπερασμού με υποθέσεις το ερώτημα $\mathcal{Q}_1$ και την πρόταση (3.1), και υποσημείωση $\Upsilon = \langle \mathcal{A}_1, \mathcal{B}_1 \rangle$, στην οποία $\mathcal{A}_1 = \{\textsf{διδάσκει}(x_1, y_1)\}$ και $\mathcal{B} = \{\textsf{διδάσκειΜεταπτυχιακό}(x_2, y_2)\}$. Επιπλέον, η σχέση $\rightsquigarrow_{\mathcal{R}} = \{\langle x_1, x_2 \rangle, \langle y_1, y_2 \rangle\}$ δηλώνει ότι κατά τον υπολογισμό του $\mathcal{R}$ η μεταβλητή x_1 μετονομάστηκε στην x_2 και η y_1 στην y_2.

Έστω τώρα, ότι μετά την ανάκτηση των αποτελεσμάτων για το $\mathcal{Q}_1$ θέλουμε να ανακτήσουμε μόνο όσους διδάσκουν κάποιο μάθημα χωρίς όμως να ανακτήσουμε και το συγκεκριμένο μάθημα, θέτουμε δηλαδή το ερώτημα

$$\mathcal{Q}'_1 = \textsf{διδάσκει}(x_1, y_1) \rightarrow \mathcal{Q}_A(x_1)$$

το οποίο αποτελεί την προβολή του $\mathcal{Q}_1$ πάνω στην πρώτη μεταβλητή x_1, δηλαδή, $\mathcal{Q}'_1 = \pi_1(\mathcal{Q}_1)$. Το σύνολο $\mathcal{R}' = \{\mathcal{Q}'_1, \mathcal{Q}'_2, \mathcal{Q}'_3\}$, όπου $\mathcal{Q}'_2 = \textsf{διδάσκειΜεταπτυχιακό}(x_2, y_2) \rightarrow \mathcal{Q}_A(x_2)$ και $\mathcal{Q}'_3 = \textsf{Καθηγητής}(x_3) \rightarrow \mathcal{Q}_A(x_3)$, αποτελεί μια επανεγγραφή για το $\mathcal{Q}'_1$ μ.β.τ. $\mathcal{T}$, με $\rightsquigarrow_{\mathcal{R}'} = \{\langle x_1, x_2 \rangle, \langle y_1, y_2 \rangle, \langle x_1, x_3 \rangle\}$.

Η επανεγγραφή $\mathcal{R}'$ μπορεί να υπολογιστεί χρησιμοποιώντας οποιοδήποτε σύγχρονο σύστημα επανεγγραφής ερωτημάτων. Παρόλα αυτά, κάθε τέτοιο σύστημα θα εφήρμοζε τους κανόνες συμπερασμού στα $\mathcal{Q}'_1$ και $\mathcal{T}$ εξαρχής χωρίς να λαμβάνει υπ' όψιν ότι ένα μεγάλο μέρος της πληροφορίας έχει ήδη παραχθεί για το ερώτημα $\mathcal{Q}_1$, και ότι το $\mathcal{Q}'_1$ είναι διαφορετικό μόνο κατά μια μεταβλητή. Πράγματι, το $\mathcal{Q}'_2$, που αποτελεί μέλος του $\mathcal{R}'$, μπορεί να παραχθεί απ' ευθείας από το $\mathcal{R}$ ως την προβολή του ερωτήματος $\mathcal{Q}_2$ στην πρώτη μεταβλητή, δηλαδή, $\mathcal{Q}'_2 = \pi_1(\mathcal{Q}_2)$. Στην πραγματικότητα, εάν τα $\mathcal{Q}_1$ και $\mathcal{Q}_2$ έχουν αποτιμηθεί σε μια βάση δεδομένων τότε μπορούμε επίσης να υπολογίσουμε απ' ευθείας την αποτίμηση των $\mathcal{Q}'_1$ και $\mathcal{Q}'_2$ χρησιμοποιώντας μια προβολή στα αποτελέσματα που επιστρέφονται για τα $\mathcal{Q}_1$ και $\mathcal{Q}_2$, αντίστοιχα.

Παρόλα αυτά, παρατηρούμε ότι το ερώτημα $\mathcal{Q}'_3$ της επανεγγραφής $\mathcal{R}'$ δεν μπορεί να υπολογιστεί χρησιμοποιώντας τον παραπάνω απλό τελεστή. Αντί αυτού, είναι απαραίτητο να χρησιμοποιηθούν οι κανόνες συμπερασμού ενός συστήματος έτσι ώστε να το υπολογίσουν. Πιο συγκεκριμένα, το $\mathcal{Q}'_3$ μπορεί να υπολογιστεί από ένα συμπερασμό, για παράδειγμα με τη χρήση συστημάτων όπως το PerfectRef ή το Requiem, με υποθέσεις το ερώτημα $\mathcal{Q}'_2$ και την πρόταση (3.2). Διαισθητικά, αυτό γίνεται επειδή μετά την εφαρμογή του τελεστή της προβολής η μεταβλητή y_2 γίνεται μη-δεσμευμένη (εμφανίζεται μόνο μια φορά), συνεπώς «ενεργοποιούνται» νέοι συμπερασμοί. $\diamondsuit$

Από το παραπάνω παράδειγμα παρατηρούμε πως εάν δίνονται ένα ερώτημα $\mathcal{Q}$ και μια επανεγγραφή $\mathcal{R}$ του $\mathcal{Q}$ με βάση κάποιο TBox $\mathcal{T}$, μια επανεγγραφή για το ερώτημα $\mathcal{Q}'$ που αποτελεί προβολή του $\mathcal{Q}$ (δηλαδή, $\mathcal{Q}' = \pi_{j_1,\ldots,j_n}(\mathcal{Q})$ για κάποιο τελεστή προβολής $\pi_{j_1,\ldots,j_n}$) μπορεί να υπολογιστεί χρησιμοποιώντας την επανεγγραφή $\mathcal{R}$ και εφαρμόζοντας δύο βασικές βασικές λειτουργίες:

Αλγόριθμος 3 RemoveVars($\mathcal{T}, \mathcal{R}, \rightsquigarrow_{\mathcal{R}}, \pi_{j_1,\ldots,j_n}$)

Είσοδος: Ένα TBox $\mathcal{T}$, μια επανεγγραφή $\mathcal{R}$ με διαμερίσεις $\mathcal{R}_D \uplus \mathcal{R}_Q$ και μια σχέση $\rightsquigarrow_{\mathcal{R}}$ που αποτελούν την έξοδο ενός αλγόριθμου επανεγγραφής για ένα ΣΕ $\mathcal{Q}$ μ.β.τ. $\mathcal{T}$, και ένας τελεστής προβολής $\pi_{j_1,\ldots,j_n}$.

1: Αρχικοποίησε μια επανεγγραφή $\mathcal{R}' := \mathcal{R}_D$ και μια σχέση $\rightsquigarrow_{\mathcal{R}'} := \rightsquigarrow_{\mathcal{R}}$
2: **for all** $\mathcal{Q}_i \in \mathcal{R}_Q$ **do**
3: Προσέθεσε το $\mathcal{Q}'_i$ στην $\mathcal{R}'$, όπου $\mathcal{Q}'_i := \pi_{j_1,\ldots,j_n}(\mathcal{Q}_i)$
4: **if** needsRewriting($\mathcal{Q}_i, \pi_{j_1,\ldots,j_n}, \mathcal{T}$) **then**
5: $\langle \mathcal{R}_n, \rightsquigarrow_{\mathcal{R}_n} \rangle :=$ rew($\mathcal{Q}'_i, \mathcal{T}$)
6: $\mathcal{R}' := \mathcal{R}' \cup \mathcal{R}_n$ και $\rightsquigarrow_{\mathcal{R}'} := \rightsquigarrow_{\mathcal{R}'} \cup \rightsquigarrow_{\mathcal{R}_n}$
7: **end if**
8: **end for**
9: **return** $\langle \mathcal{R}', \rightsquigarrow_{\mathcal{R}'} \rangle$

1. προβολή όλων των ερωτημάτων από το $\mathcal{R}$ χρησιμοποιώντας τον τελεστή $\pi_{j_1,\ldots,j_n}$, και στη συνέχεια,

2. πιθανώς εφαρμογή επιπλέον συμπερασμών πάνω στα ερωτήματα που έχουν προκύψει έτσι ώστε να υπολογιστούν αυτά που υπολείπονται.

Γενικά, είναι δυνατό να εφαρμόσουμε τους κανόνες συμπερασμού ενός συστήματος επανεγγραφής εξαντλητικά σε όλα τα προβληθέντα ερωτήματα. Κάτι τέτοιο όμως, μπορεί ενδεχομένως να επηρεάσει την επίδοση του αλγορίθμου. Αντιθέτως ο αλγόριθμος μας ελέγχει εάν έχει ενεργοποιηθεί ένας νέος συμπερασμός χρησιμοποιώντας την παρακάτω συνάρτηση.

Ορισμός 3.1.2. Έστω $\mathcal{T}$ TBox, έστω $\mathcal{Q}$ ΣΕ, και έστω $\pi_{j_1,\ldots,j_n}$ τελεστής προβολής. Τότε η συνάρτηση needsRewriting($\mathcal{Q}, \pi_{j_1,\ldots,j_n}, \mathcal{T}$) επιστρέφει αληθές εάν υπάρχει συμπερασμός με το ερώτημα $\pi_{j_1,\ldots,j_n}(\mathcal{Q})$ ως κύρια υπόθεση και υποσημειώσεις σ_π, Υ_π και υπάρχει $\langle \mathcal{A}_\pi, \mathcal{B}_\pi \rangle \in \Upsilon_\pi$ τέτοια ώστε για κάθε συμπερασμό με το ερώτημα $\mathcal{Q}$ ως κύρια υπόθεση και κάθε $\langle \mathcal{A}, \mathcal{B} \rangle \in \Upsilon$ είτε $\mathcal{A} \neq \mathcal{A}_\pi$ ή $\mathcal{B} \neq \mathcal{B}_\pi$, διαφορετικά επιστρέφει ψευδές. △

Ο αλγόριθμος μας για τον υπολογισμό μιας επανεγγραφής για ερωτήματα που έχουν προκύψει με χρήση του τελεστή της προβολής χρησιμοποιώντας μια επανεγγραφή που έχει υπολογιστεί προηγούμενα φαίνεται στον Αλγόριθμο 3. Ο αλγόριθμος δέχεται ως είσοδο ένα TBox $\mathcal{T}$, την επανεγγραφή $\mathcal{R}$ για ένα ερώτημα $\mathcal{Q}$ μ.β.τ. $\mathcal{T}$ που έχει υπολογιστεί νωρίτερα, τη σχέση $\rightsquigarrow_{\mathcal{R}}$, και έναν τελεστή προβολής για το ερώτημα $\mathcal{Q}$. Εσωτερικά, χρησιμοποιεί επίσης έναν αλγόριθμο επανεγγραφής rew έτσι ώστε να μπορεί να εφαρμόζει τον κανόνα συμπερασμού όπου αυτό κρίνεται απαραίτητο. Στην αρχή, ο αλγόριθμος αρχικοποιεί μια νέα επανεγγραφή $\mathcal{R}'$ με το σύνολο $\mathcal{R}_D$ και τη σχέση $\rightsquigarrow_{\mathcal{R}'}$ με την $\rightsquigarrow_{\mathcal{R}}$. Στη συνέχεια, προσπελαύνει όλα τα ερωτήματα $\mathcal{Q}_i$ στο $\mathcal{R}$ και υπολογίζει τις προβολές $\pi_{j_1,\ldots,j_n}(\mathcal{Q}_i)$ προσθέτοντας τις στο αποτέλεσμα. Επιπλέον, για κάθε ερώτημα στο οποίο έχει εφαρμοστεί ο τελεστής της προβολής ελέγχει εάν χρειάζεται επιπλέον επανεγγραφή με τη χρήση της συνάρτησης needsRewriting και

εάν κάτι τέτοιο είναι απαραίτητο καλεί τον αλγόριθμο rew με εισόδους το ερώτημα $\pi_{j_1,\ldots,j_n}(\mathcal{Q}_i)$ και το TBox $\mathcal{T}$.

Θεώρημα 3.1.3. *Έστω $\mathcal{Q}$ ΣΕ, έστω $\mathcal{T}$ TBox, έστω $\pi_{j_1,\ldots,j_n}$ ένας τελεστής προβολής στο $\mathcal{Q}$, και έστω $\mathcal{R}$ και $\rightsquigarrow_{\mathcal{R}}$ η έξοδος ενός αλγόριθμου επανεγγραφής όταν εφαρμοστεί στα $\mathcal{Q}$ και $\mathcal{T}$. Όταν τα $\mathcal{T}$, $\mathcal{R}$, $\rightsquigarrow_{\mathcal{R}}$, και $\pi_{j_1,\ldots,j_n}$ δίνονται ως είσοδος ο Αλγόριθμος 3 τερματίζει. Έστω $\langle \mathcal{R}', \rightsquigarrow_{\mathcal{R}'} \rangle$ η έξοδος του αλγορίθμου, τότε το σύνολο $\mathcal{R}'$ είναι μια επανεγγραφή για το $\pi_{j_1,\ldots,j_n}(\mathcal{Q})$ μ.β.τ. $\mathcal{T}$.*

Απόδειξη: Ο τερματισμός προκύπτει από το γεγονός ότι ο Αλγόριθμος 3 προσπελαύνει κάθε μέλος ενός πεπερασμένου συνόλου ($\mathcal{R}_Q$) μόνο μια φορά και επειδή ο αλγόριθμος επανεγγραφής rew τερματίζει.

Έστω, $\mathcal{T}, \langle \mathcal{R}, \rightsquigarrow_{\mathcal{R}} \rangle, \mathcal{Q}$, και $\pi_{j_1,\ldots,j_n}$ οι είσοδοι του Αλγόριθμου 3, όπου η πλειάδα $\langle \mathcal{R}, \rightsquigarrow_{\mathcal{R}} \rangle$ υπολογίζεται από κάποιο αλγόριθμο επανεγγραφής rew. Υποθέτουμε τώρα ότι η έξοδος του Αλγόριθμου 3 είναι το ζευγάρι $\langle \mathcal{R}', \rightsquigarrow_{\mathcal{R}'} \rangle$ καθώς και ότι το ζευγάρι $\langle \mathcal{R}_i, \rightsquigarrow_{\mathcal{R}_i} \rangle$ είναι η επανεγγραφή που παράγεται ύστερα από i βήματα του αλγόριθμου rew όταν αυτός εφαρμόζεται στα $\mathcal{Q}'$ και $\mathcal{T}$.

Η πληρότητα του αλγορίθμου στηρίζεται στην παρακάτω ιδιότητα, την οποία δείχνουμε με χρήση επαγωγής πάνω στο i:

> (VR): Για κάθε $i \geq 0$ και για όλα τα $\mathcal{Q}'_l \in \mathcal{R}_i$ υπάρχει $\mathcal{Q}_l \in \mathcal{R}$ τέτοιο ώστε για το $\mathcal{Q}^{\pi}_l = \pi_{j_1,\ldots,j_n}(\mathcal{Q}_l)$ ισχύει μια από τις παρακάτω συνθήκες:
>
> 1. το $\mathcal{Q}^{\pi}_l$ υπάγει το $\mathcal{Q}'_l$, ή
> 2. η συνάρτηση needsRewriting$(\mathcal{Q}_l, \pi_{j_1,\ldots,j_n}, \mathcal{T})$ επιστρέφει αληθές και κάποιο ερώτημα $\mathcal{Q}''_l \in$ rew$(\mathcal{Q}^{\pi}_l, \mathcal{T})$ υπάγει το $\mathcal{Q}'_l$.

Βασική Περίπτωση ($i = 0$): Αρχικά $\mathcal{R}_0 = \{\mathcal{Q}'\}$ όπου $\mathcal{Q}' = \pi_{j_1,\ldots,j_n}(\mathcal{Q})$, συνεπώς η Ιδιότητα (VR) ικανοποιείται.

Βήμα Επαγωγής: Υποθέτουμε ότι η Ιδιότητα (VR) ισχύει για το βήμα i για κάθε $\mathcal{Q}'_l \in \mathcal{R}_i$—δηλαδή, είτε το 1. είτε το 2. ισχύουν από την Ιδιότητα (VR). Στη συνέχεια, υποθέτουμε ότι στο βήμα $i+1$ ο αλγόριθμος rew εφαρμόζει ένα βήμα συμπερασμού σε κάποιο ερώτημα $\mathcal{Q}'_l$ παράγοντας το ερώτημα $\mathcal{Q}'_{l+1}$ που δεν περιέχει συναρτησιακά σύμβολα. Εξετάζουμε τώρα ξεχωριστά τις δύο περιπτώσεις της επαγωγικής υπόθεσης:

1. Το $\mathcal{Q}^{\pi}_l$ υπάγει το $\mathcal{Q}'_l$. Από τις ιδιότητες της υπαγωγής θα ισχύει ότι είτε το $\mathcal{Q}^{\pi}_l$ υπάγει το $\mathcal{Q}'_{l+1}$ ή ένας συμπερασμός είναι εφαρμόσιμος στο $\mathcal{Q}^{\pi}_l$ και το αποτέλεσμα, έστω $\mathcal{Q}^{\pi}_0$, υπάγει το $\mathcal{Q}'_{l+1}$. Εάν η πρώτη περίπτωση είναι αληθής τότε όντως υπάρχει ερώτημα $\mathcal{Q}_l \in \mathcal{R}$ τ.ω. το $\mathcal{Q}^{\pi}_l$ να υπάγει το $\mathcal{Q}'_{l+1}$. Υποθέτουμε ότι ισχύει η δεύτερη περίπτωση. Υπενθυμίζουμε ότι το $\mathcal{Q}_l$ περιέχει τα ίδια άτομα σώματος με το $\mathcal{Q}^{\pi}_l$. Εφόσον οι αλγόριθμοι επανεγγραφής πραγματοποιούν συμπερασμούς χρησιμοποιώντας μόνο τα άτομα του σώματος των ερωτημάτων, προκύπτει ότι ο ίδιος συμπερασμός μπορεί να εφαρμοστεί στο $\mathcal{Q}_l$ παράγοντας το $\mathcal{Q}_0$. Παρόλα αυτά, εφόσον το $\mathcal{Q}_l$ περιέχει περισσότερες διακεκριμένες μεταβλητές από το $\mathcal{Q}^{\pi}_l$, το $\mathcal{Q}_0$ μπορεί να περιέχει συναρτησιακούς όρους στις μεταβλητές αυτές στο άτομο της κεφαλής οι οποίες όμως έχουν απαλειφθεί από την προβολή

στο $\mathcal{Q}_l^\pi$. Εάν δεν περιέχει, τότε $\mathcal{Q}_0 \in \mathcal{R}$ και συνεπώς $\mathcal{Q}_0^\pi = \pi_{j_1,\ldots,j_n}(\mathcal{Q}_0)$ καθώς η μόνη διαφορά ανάμεσα στα $\mathcal{Q}_0$ και $\mathcal{Q}_0^\pi$ είναι οι διακεκριμένες μεταβλητές οι οποίες έχουν απαλειφθεί από την προβολή. Εάν περιέχει συναρτησιακούς όρους στην κεφαλή, τότε θα ισχύει ότι $\mathcal{Q}_0 \notin \mathcal{R}$ εφόσον ένας αλγόριθμος Q-προσανατολισμένος παράγει μόνο ερωτήματα που δεν περιέχουν συναρτησιακούς όρους. Τότε όμως, η συνάρτηση needsRewriting$(\mathcal{Q}_l, \pi_{j_1,\ldots,j_n}, \mathcal{T})$ επιστρέφει αληθές και είναι προφανές ότι $\mathcal{Q}_0^\pi \in$ rew$(\mathcal{Q}_l^\pi, \mathcal{T})$. Συνεπώς, σε κάθε περίπτωση η Ιδιότητα (VR) ισχύει.

2. Η needsRewriting$(\mathcal{Q}_l, \pi_{j_1,\ldots,j_n}, \mathcal{T})$ επιστρέφει αληθές και υπάρχει ερώτημα $\mathcal{Q}_l'' \in$ rew$(\mathcal{Q}_l^\pi, \mathcal{T})$ που υπάγει το $\mathcal{Q}_l'$. Και πάλι, είτε το $\mathcal{Q}_l''$ υπάγει το $\mathcal{Q}_{l+1}'$ ή μπορεί να εφαρμοστεί ένας συμπερασμός στο $\mathcal{Q}_l''$ και το αποτέλεσμα, έστω $\mathcal{Q}_0''$ υπάγει το $\mathcal{Q}_l''$. Σε κάθε περίπτωση το 2. θα ισχύει καθώς θα έχουμε είτε $\mathcal{Q}_l'' \in$ rew$(\mathcal{Q}_l^\pi, \mathcal{T})$ ή $\mathcal{Q}_0'' \in$ rew$(\mathcal{Q}_l^\pi, \mathcal{T})$.

Όλα αυτά αποδεικνύουν την Ιδιότητα (VR).

Υποθέτουμε τώρα ότι όταν εφαρμόζεται για τα $\mathcal{Q}', \mathcal{T}$, ο αλγόριθμος επανεγγραφής rew τερματίζει ύστερα από n βήματα έχοντας υπολογίσει την επανεγγραφή $\mathcal{R}_n$. Η Ιδιότητα (VR) υποδηλώνει ότι με τον τερματισμό του Αλγορίθμου 3 για κάθε ερώτημα $\mathcal{Q}_l \in \mathcal{R}_n$ θα υπάρχει ένα ερώτημα $\mathcal{Q}' \in \mathcal{R}'$ το οποίο υπάγει το $\mathcal{Q}_l$. □

Ένα σημείο του Αλγορίθμου 3 που είναι υπολογιστικά ακριβό είναι στη γραμμή 5 όπου καλείται ο αλγόριθμος επανεγγραφής rew. Εάν αυτός καλείται αρκετές φορές τότε η απόδοση του αλγορίθμου μας μπορεί να επηρεαστεί σημαντικά. Συνεπώς για να είναι αποδοτικός ο Αλγόριθμος 3 είναι απαραίτητο ο rew να καλείται όσο το δυνατόν λιγότερες φορές. Μια προσέγγιση που έχει χρησιμοποιηθεί στο πρωτότυπο σύστημα που θα παρουσιασουμε στο κεφάλαιο 6 είναι η ακόλουθη. Έστω $\mathcal{Q}_i'$ η προβολή ενός ερωτήματος $\mathcal{Q}_i$ και έστω ότι το $\mathcal{Q}_i$ ικανοποιεί τις συνθήκες της γραμμής 4. Στην περίπτωση αυτή αντί να επανεγγράψουμε άμεσα το ερώτημα $\mathcal{Q}_i'$ το προσθέτουμε σε ένα σύνολο toRew. Τότε, όταν τερματίσει ο εξωτερικός βρόχος επανάληψης (γραμμές 2–8), αφαιρούμε από το toRew όλα τα ερωτήματα που είναι περιττά σε αυτό και συνεπώς εκτελούμε τον rew μόνο για τα μη-περιττά ερωτήματα. Επιπρόσθετα, μια επανεγγραφή που έχει ήδη υπολογιστεί για κάποιο ερώτημα $\mathcal{Q}_i \in$ toRew μπορεί να χρησιμοποιηθεί για να αφαιρεθούν ερωτήματα από το toRew για τα οποία δεν έχει εκτελεστεί ακόμα ο rew. Τέλος, μια άλλη ευριστική μέθοδος είναι να ελέγξουμε εάν η προβολή του ερωτήματος $\mathcal{Q}$ για το οποίο έχει υπολογιστεί η $\mathcal{R}$ βρίσκεται στο toRew. Στην περίπτωση αυτή ο αλγόριθμος μπορεί να σταματήσει την εκτέλεση του και να εκτελέσει τον rew μόνο για το $\pi_{j_1,\ldots,j_n}(\mathcal{Q})$ καθώς κάτι τέτοιο θα παράγει την επιθυμητή επανεγγραφή.

3.2 Επέκταση Ερωτημάτων με Διακεκριμένες Μεταβλητές

Στην ενότητα αυτή μελετάμε το πρόβλημα του υπολογισμού μιας επανεγγραφής για ένα ερώτημα του οποίου οι διακεκριμένες μεταβλητές (μεταβλητές απάντησης) έχουν επεκταθεί με νέες, εκμεταλλευόμενοι και πάλι όσο το δυνατόν περισσότερη

πληροφορία από την επανεγγραφή που έχει ήδη υπολογιστεί για το αρχικό ερώτημα. Το παρακάτω παράδειγμα περιγράφει τις βασικές ιδέες στις οποίες στηρίζεται ο αλγόριθμος που θα παρουσιάσουμε στη συνέχεια.

Παράδειγμα 3.2.1. Έστω το TBox $\mathcal{T}$, το ερώτημα $\mathcal{Q}'_1$, η επανεγγραφή $\mathcal{R}' = \{\mathcal{Q}'_1, \mathcal{Q}'_2, \mathcal{Q}'_3\}$ και η σχέση $\rightsquigarrow_{\mathcal{R}'}$ από το Παράδειγμα 3.1.1. Υποθέτουμε τώρα ότι θέλουμε να «επεκτείνουμε» το ερώτημα αυτό και να ανακτήσουμε επιπλέον και τα αντίστοιχα μαθήματα, δηλαδή, θέλουμε να υπολογίσουμε την επανεγγραφή του ερωτήματος $\mathcal{Q}_1$ από το Παράδειγμα 3.1.1 η οποία όπως δείξαμε προηγουμένως αποτελείται από το σύνολο $\mathcal{R} = \{\mathcal{Q}_1, \mathcal{Q}_2\}$ που μπορεί να υπολογιστεί από τα $\mathcal{Q}_1$ και $\mathcal{T}$ χρησιμοποιώντας έναν οποιοδήποτε αλγόριθμο επανεγγραφής.

Παρόλα αυτά, αξίζει να σημειώσουμε ότι και πάλι αντί να εκτελέσουμε έναν αλγόριθμο επανεγγραφής εξαρχής μπορούμε να εκμεταλλευτούμε την προηγουμένως υπολογισμένη επανεγγραφή $\mathcal{R}'$ του $\mathcal{Q}'_1$ μ.β.τ. $\mathcal{T}$. Πιο συγκεκριμένα, το $\mathcal{Q}_1$ μπορεί να υπολογιστεί από το $\mathcal{Q}'_1$ αντικαθιστώντας το άτομο κεφαλής $Q_A(x_1)$ του $\mathcal{Q}'_1$ με το άτομο $Q_A(x_1, y_1)$ (η y_1 είναι η νέα διακεκριμένη μεταβλητή που χρησιμοποιείται για να ανακτήσουμε τα μαθήματα που διδάσκονται). Επιπλέον, το $\mathcal{Q}'_2$ μπορεί να υπολογιστεί αντικαθιστώντας το άτομο κεφαλής $Q_A(x_2)$ του $\mathcal{Q}'_2$ με το άτομο $Q_A(x_1, y_1)\tau$, όπου η αντικατάσταση $\tau = \{x_1 \mapsto x_2, y_1 \mapsto y_2\}$ υπολογίζεται από την σχέση $\rightsquigarrow_{\mathcal{R}'}$. Τέλος, παρατηρούμε ότι το ερώτημα $\mathsf{Καθηγητής}(x_3) \rightarrow Q_A(x_3, y_1)$ που παράγεται από το $\mathcal{Q}_3$ χρησιμοποιώντας την παραπάνω διαδικασία δεν προκύπτει από το $\mathcal{T} \cup \mathcal{Q}$, συνεπώς δεν πρέπει να παραχθεί κανένα ερώτημα από το $\mathcal{Q}_3$. $\diamondsuit$

Το παραπάνω παράδειγμα υποδηλώνει ότι δοθέντος ενός ερωτήματος $\mathcal{Q}$ που έχει άτομο κεφαλής το $Q_A(\vec{x})$, μιας επανεγγραφής $\mathcal{R}$ και σχέσης $\rightsquigarrow_{\mathcal{R}}$ για το $\mathcal{Q}$ με βάση κάποιο TBox $\mathcal{T}$ και ένα νέο διάνυσμα $\vec{y}$ που περιέχει διακεκριμένες μεταβλητές τ.ω. $\vec{y} \subseteq \mathsf{var}(\mathcal{Q})$, ο αλγόριθμος μας για να υπολογίσει μια επανεγγραφη για το ερώτημα $\bigwedge \mathsf{bd}(\mathcal{Q}) \rightarrow Q_A(\vec{x}, \vec{y})$ χρειάζεται να εργαστεί ως ακολούθως: για κάθε ερώτημα $\mathcal{Q}_i \in \mathcal{R}$ εάν $\vec{y}\tau \subseteq \mathsf{var}(\mathcal{Q}_i)$, όπου η αντικατάσταση τ κατασκευάζεται από τη σχέση $\rightsquigarrow_{\mathcal{R}}$, τότε επεκτείνει τις διακεκριμένες μεταβλητές του $\mathcal{Q}_i$ προσθέτοντας τις $\vec{y}\tau$. Διαφορετικά θα πρέπει να απορρίψει το $\mathcal{Q}_i$. Η διαδικασία αυτή γίνεται σαφής στον παρακάτω ορισμό.

Ορισμός 3.2.2. Έστω $\mathcal{T}$ TBox, έστω $\mathcal{Q}$ ΣΕ με διακεκριμένες μεταβλητές $\vec{x}$, και έστω $\langle \mathcal{R}_D \uplus \mathcal{R}_Q, \rightsquigarrow_{\mathcal{R}} \rangle$ έξοδος ενός αλγορίθμου επανεγγραφής όταν εφαρμόζεται για τα $\mathcal{Q}, \mathcal{T}$. Επιπλέον, έστω $\vec{z}$ διάνυσμα μεταβλητών τ.ω. $\vec{z} \subseteq \mathsf{var}(\mathcal{Q})$, έστω $\mathcal{Q}' \in \mathcal{R}_Q$, και έστω $\tau := \{x \mapsto y \mid x \in \mathsf{var}(\mathcal{Q}), y \in \mathsf{var}(\mathcal{Q}')$ και $x \rightsquigarrow_{\mathcal{R}} y\}$. Η *επέκταση* του $\mathcal{Q}'$ με το $\vec{z}$ είναι η πρόταση $\mathcal{Q}''$ που ορίζεται ως εξής:

$$\mathcal{Q}'' = \bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow Q_A(\vec{x}, \vec{z})\tau \tag{3.3}$$

Η επέκταση ονομάζεται *ασφαλής* (*safe*) εάν $\vec{z}\tau \subseteq \mathsf{var}(\mathcal{Q}')$. $\triangle$

Στο Παράδειγμα 3.2.1 υπολογίσαμε την επανεγγραφή του ερωτήματος $\mathcal{Q}_1$ που είναι η επέκταση του $\mathcal{Q}'_1$ υπολογίζοντας όλες τις ασφαλείς επεκτάσεις των ερωτημάτων από την επανεγγραφή $\mathcal{R}'$ χωρίς να χρειαστεί να εφαρμόσουμε συμπερασμούς. Κάτι τέτοιο, όμως, δεν είναι πάντα δυνατό, και όπως δείχνει και το παρακάτω παράδειγμα εξαρτάται σε μεγάλο βαθμό από την επανεγγραφή εισόδου.

Παράδειγμα 3.2.3. Έστω το παρακάτω TBox και ερώτημα:

$$\mathcal{T} = \{A(x) \rightarrow \exists y.R(x,y)\} \qquad \mathcal{Q}_1 = A(x_1) \wedge R(x_1,y_1) \rightarrow \mathcal{Q}_A(x_1).$$

Το σύνολο $\mathcal{R} = \{\mathcal{Q}_1, \mathcal{Q}_2\}$, όπου $\mathcal{Q}_2 = A(x_2) \rightarrow \mathcal{Q}_A(x_2)$, είναι μια επανεγγραφή του $\mathcal{Q}_1$ μ.β.τ. $\mathcal{T}$. Παρόλα αυτά, παρατηρούμε ότι το ερώτημα $\mathcal{Q}_2$ υπάγει το $\mathcal{Q}_1$, συνεπώς το $\mathcal{Q}_1$ μπορεί να αφαιρεθεί και το σύνολο $\mathcal{R}' = \{\mathcal{Q}_2\}$ είναι επίσης μια επανεγγραφή του $\mathcal{Q}_1$ μ.β.τ. $\mathcal{T}$.

Έστω τώρα ότι θέλουμε να επεκτείνουμε το ερώτημα $\mathcal{Q}_1$ προσθέτοντας την μεταβλητή y_1 στη λίστα των διακεκριμένων μεταβλητών και ότι θέλουμε να υπολογίσουμε μια επανεγγραφή για το επεκτεταμένο ερώτημα, δηλαδή, το $\mathcal{Q}_1' = A(x_1) \wedge R(x_1,y_1) \rightarrow \mathcal{Q}_A(x_1,y_1)$. Χρησιμοποιώντας την $\mathcal{R}'$ το μόνο ερώτημα που μπορεί να επεκταθεί με την y_1 είναι το $\mathcal{Q}_2$, όμως, η επέκταση του δεν είναι ασφαλής. Συνεπώς, χρησιμοποιώντας την $\mathcal{R}'$ προκύπτει το άδειο σύνολο. Αντιθέτως χρησιμοποιώντας την $\mathcal{R}$ η επέκταση του $\mathcal{Q}_1$ με την y_1 είναι το ερώτημα $\mathcal{Q}_1'$ και το σύνολο $\{\mathcal{Q}_1'\}$ η επιθυμητή επανεγγραφή. ♢

Διαισθητικά, το πρόβλημα με το προηγούμενο παράδειγμα είναι ότι ενώ το $\mathcal{Q}_2$ υπάγει το $\mathcal{Q}_1$ στην $\mathcal{R}$, οι επεκτάσεις του ερωτήματος $\mathcal{Q}_2$ με κάποια μεταβλητή μπορεί να μην είναι ασφαλείς και συνεπώς δεν αποτελούν μέρος της εξόδου. Συνεπώς, οι επεκτάσεις του $\mathcal{Q}_1$ δεν θα είναι περιττές και θα πρέπει να προστεθούν στο αποτέλεσμα. Κάτι τέτοιο είναι δυνατό εάν το $\mathcal{Q}_1$ δεν έχει αφαιρεθεί από την $\mathcal{R}$ λόγω του $\mathcal{Q}_2$. Εάν αφαιρούνται περιττά ερωτήματα τότε ο αλγόριθμος μας θα πρέπει να εφαρμόζει επιπλέον συμπερασμούς στα επεκτεταμένα ερωτήματα καθώς και στην επέκταση του αρχικού ερωτήματος $\mathcal{Q}$. Όμως, αυτό γενικά συνεπάγεται ότι μια διαδικασία επανεγγραφής θα πρέπει να εφαρμοστεί εξαρχής ακυρώνοντας έτσι όποια οφέλη προέκυψαν από την πληροφορία που έχει υπολογιστεί προηγουμένως. Στη συνέχεια, ορίζουμε μια ιδιότητα για τις επανεγγραφές που χρησιμοποιούνται ως είσοδοι η οποία είναι επαρκής για τον υπολογισμό μιας επανεγγραφής χρησιμοποιώντας μόνο ασφαλείς επεκτάσεις.

Ορισμός 3.2.4. Έστω $\mathcal{Q}$ ΣΕ, έστω $\mathcal{T}$ TBox, και έστω $\mathcal{R}$ επανεγγραφή για το $\mathcal{Q}$ μ.β.τ. $\mathcal{T}$ με διαμερίσεις $\mathcal{R}_D \uplus \mathcal{R}_Q$ υπολογισμένη από ένα αλγόριθμο επανεγγραφής που στηρίζεται σε ένα σύστημα συμπερασμών Γ. Λέμε ότι η $\mathcal{R}$ είναι *κλειστή ως προς το συμπερασμό* (*inference-closed*) για τα $\mathcal{Q}, \mathcal{T}$ εάν $\mathcal{Q} \in \mathcal{R}$ και εάν επιπλέον, για κάθε ερώτημα $\mathcal{Q}_1 \in \mathcal{R}_Q$ εάν ένα ερώτημα $\mathcal{Q}_2$ προκύπτει από το $\mathcal{T} \cup \mathcal{Q}_1$ μέσω του Γ, τότε επίσης $\mathcal{Q}_2 \in \mathcal{R}_Q$. △

Εύκολα παρατηρούμε ότι η επανεγγραφή $\mathcal{R}'$ από το Παράδειγμα 3.2.3 δεν είναι κλειστή ως προς το συμπερασμό καθώς $\mathcal{Q}_1 \notin \mathcal{R}'$, ενώ η $\mathcal{R}$ είναι.

Διαισθητικά, μια επανεγγραφή $\mathcal{R}$ είναι κλειστή ως προς το συμπερασμό εάν όλα τα ερωτήματα πο παράγονται από τον αλγόριθμο κατά την παραγωγή της $\mathcal{R}$ ανήκουν επίσης στην $\mathcal{R}$. Αυτό συνήθως συνεπάγεται ότι η $\mathcal{R}$ υπολογίζεται χωρίς τη χρήση βελτιστοποιήσεων όπως η απαλοιφή περιττών ερωτημάτων που αφαιρεί ερωτήματα που προκύπτουν από το Γ, ή ότι τέτοιες βελτιστοποιήσεις έχουν απενεργοποιηθεί. Παρόλο που κάτι τέτοιο συνεπάγεται ότι η παραγωγή μιας επανεγγραφής, που χρησιμοποιείται στη συνέχεια ως είσοδος στον αλγόριθμο μας, δεν είναι αποδοτική υπάρχουν συστήματα όπως θα δείξουμε στη συνέχεια της διατριβής μας που μπορούν να υπολογίσουν

Αλγόριθμος 4 ExtendRewritingForNewVars($\mathcal{Q}, \mathcal{R}, \rightsquigarrow_{\mathcal{R}}, \vec{y}$)

Είσοδος: Ένα ΣΕ $\mathcal{Q}$, μια επανεγγραφή $\mathcal{R}$ με διαμέριση $\mathcal{R}_D \uplus \mathcal{R}_Q$ και μια σχέση $\rightsquigarrow_{\mathcal{R}}$ που αποτελούν την έξοδο ενός αλγορίθμου επανεγγραφής για το $\mathcal{Q}$ με βάση ένα TBox $\mathcal{T}$, και ένα διάνυσμα μεταβλητών $\vec{y}$ τ.ω. $\vec{y} \subseteq \mathsf{var}(\mathcal{Q})$.

1: Αρχικοποίησε μια επανεγγραφή $\mathcal{R}' := \mathcal{R}_D$ και μια σχέση $\rightsquigarrow_{\mathcal{R}'} := \rightsquigarrow_{\mathcal{R}}$
2: **for all** $\mathcal{Q}_i \in \mathcal{R}_Q$ **do**
3: το $\mathcal{Q}'_i :=$ είναι η επέκταση του $\mathcal{Q}_i$ με το $\vec{y}$
4: **if** το $\mathcal{Q}'_i$ είναι ασφαλές **then**
5: Προσέθεσε το $\mathcal{Q}'_i$ στην $\mathcal{R}'$
6: **end if**
7: **end for**
8: **return** $\langle \mathcal{R}', \rightsquigarrow_{\mathcal{R}'} \rangle$

τέτοιες επανεγγραφές πολύ αποδοτικά. Επιπρόσθετα, η ιδιότητα της κλειστότητας ως προς τον συμπερασμό φαίνεται πως είναι σχετική και με άλλα προβλήματα στην απάντηση ερωτημάτων, όπως η επανεγγραφή ερωτημάτων για οντολογίες που έχουν τροποποιηθεί [161].

Ο Αλγόριθμος 4 παρουσιάζει την προσέγγιση μας για τον υπολογισμό μιας επανεγγραφής για ερωτήματα που έχουν επεκταθεί με νέες διακεκριμένες μεταβλητές συνοψίζοντας όσα παρουσιάσαμε παραπάνω. Δέχεται ως είσοδο ένα ερώτημα $\mathcal{Q}$, μια επανεγγραφή $\langle \mathcal{R}, \rightsquigarrow_{\mathcal{R}} \rangle$ για το $\mathcal{Q}$ μ.β.τ. $\mathcal{T}$, και ένα διάνυσμα μεταβλητών $\vec{y}$. Στη συνέχεια προσπελαύνει όλα τα ΣΕ $\mathcal{Q}_i \in \mathcal{R}_Q$ και υπολογίζει την επέκταση $\mathcal{Q}'_i$ του $\mathcal{Q}_i$ με το διάνυσμα $\vec{y}$ όπως στον Ορισμό 3.2.2. Εάν η επέκταση είναι ασφαλής τότε το $\mathcal{Q}'_i$ προστίθεται στην $\mathcal{R}'$ (γραμμή 5), αλλιώς το $\mathcal{Q}'_i$ απορρίπτεται.

Θεώρημα 3.2.5. *Έστω $\mathcal{T}$ TBox, έστω $\mathcal{Q}$ ΣΕ, έστω $\vec{y}$ διάνυσμα μεταβλητών τ.ω. $\vec{y} \subseteq \mathsf{var}(\mathcal{Q})$ και έστω $\mathcal{R}$ μια επανεγγραφή κλειστή ως προς το συμπερασμό για το $\mathcal{Q}$ μ.β.τ. $\mathcal{T}$ μαζί με μια σχέση $\rightsquigarrow_{\mathcal{R}}$. Όταν εφαρμόζεται ο Αλγόριθμος 4 στα $\mathcal{Q}, \mathcal{R}, \rightsquigarrow_{\mathcal{R}}$, και $\vec{y}$ τερματίζει. Έστω $\mathcal{R}'$ η έξοδος του αλγορίθμου. Τότε η $\mathcal{R}'$ είναι μια επανεγγραφή κλειστή ως προς το συμπερασμό για την επέκταση του $\mathcal{Q}$ με το διάνυσμα $\vec{y}$ μ.β.τ. $\mathcal{T}$.*

Απόδειξη: Ο τερματισμός προκύπτει από το γεγονός ότι ο Αλγόριθμος 4 προσπελαύνει μια φορά κάθε μέλος του πεπερασμένου συνόλου $\mathcal{R}$.

Έστω, $\mathcal{T}$, $\langle \mathcal{R}, \rightsquigarrow_{\mathcal{R}} \rangle$, $\mathcal{Q}$, και $\vec{y}$, η είσοδος του Αλγόριθμου 4, όπου η πλειάδα $\langle \mathcal{R}, \rightsquigarrow_{\mathcal{R}} \rangle$ υπολογίζεται από κάποιο αλγόριθμο επανεγγραφής rew και το $\vec{y}$ είναι ένα διάνυσμα μεταβλητών τ.ω. $\vec{y} \subseteq \mathsf{var}(\mathcal{Q})$.

Έστω τώρα ότι η έξοδος του Αλγορίθμου 4 είναι το ζευγάρι $\langle \mathcal{R}', \rightsquigarrow_{\mathcal{R}'} \rangle$ και έστω επίσης ότι το ζευγάρι $\langle \mathcal{R}_i, \rightsquigarrow_{\mathcal{R}_i} \rangle$ είναι η επανεγγραφή που προκύπτει ύστερα από i βήματα του αλγόριθμου επανεγγραφής rew όταν αυτός εφαρμόζεται στην επέκταση του $\mathcal{Q}$ με το $\vec{y}$, το οποίο στη συνέχεια θα αναφέρεται ως $\mathcal{Q}'$.

Η απόδειξη του παραπάνω θεωρήματος προκύπτει από την παρακάτω ιδιότητα, την οποία δείχνουμε με χρήση επαγωγής στο βήμα i:

(VE): Για κάθε $i \geq 0$ και για κάθε $\mathcal{Q}'_l \in \mathcal{R}_i$ υπάρχει $\mathcal{Q}_l \in \mathcal{R}$ τ.ω. η επέκταση $\mathcal{Q}_e$ του $\mathcal{Q}_l$ με το $\vec{y}$ είναι ασφαλής και υπάγει το $\mathcal{Q}'_l$.

Βασική Περίπτωση ($i = 0$): Αρχικά $\mathcal{R}_0 = \{\mathcal{Q}'\}$ όπου $\mathcal{Q}'$ είναι η επέκταση του $\mathcal{Q}$ με το $\vec{y}$, συνεπώς η Ιδιότητα (VE) ικανοποιείται.

Βήμα Επαγωγής: Έστω ότι η Ιδιότητα (VE) ισχύει για το βήμα i για κάθε $\mathcal{Q}'_l \in \mathcal{R}_i$. Επιπλέον, έστω ότι στο βήμα $i+1$ παράγεται το ερώτημα $\mathcal{Q}'_{l+1}$ από το $\mathcal{Q}'_l$ με την εφαρμογή ενός βήματος συμπερασμού ενός Q-προσανατολισμένου αλγόριθμου επανεγγραφής rew. Εφόσον το $\mathcal{Q}_e$ υπάγει το $\mathcal{Q}'_l$ από τις ιδιότητες της υπαγωγής θα έχουμε ότι είτε το $\mathcal{Q}_e$ υπάγει το $\mathcal{Q}'_{l+1}$ ή ότι ένας συμπερασμός μπορεί να εφαρμοστεί στο $\mathcal{Q}_e$ και το αποτέλεσμα $\mathcal{Q}'_e$ υπάγει το $\mathcal{Q}'_{l+1}$. Εάν ισχύει η πρώτη περίπτωση τότε η ιδιότητα ισχύει. Συνεπώς θεωρούμε ότι ισχύει η δεύτερη περίπτωση. Εφόσον όλες οι μεταβλητές στο διάνυσμα $\vec{y}$ εμφανίζονται στην κεφαλή του $\mathcal{Q}_e$ ο συμπερασμός που παράγει το $\mathcal{Q}'_e$ πρέπει να είναι τέτοιος ώστε να μην απαλείφει ή αντιστοιχεί μεταβλητές από το $\vec{y}$ σε συναρτησιακούς όρους. Αξίζει τώρα να υπενθυμίσουμε ότι το $\mathcal{Q}_l$ είναι όπως το $\mathcal{Q}_e$, αλλά το δεύτερο περιέχει όλες τις μεταβλητές $\vec{y}$ στην κεφαλή του. Συνεπώς, οι ίδιοι συμπερασμοί είναι εφαρμόσιμοι στο $\mathcal{Q}_l$, παράγοντας το $\mathcal{Q}_0$. Επιπρόσθετα, όπως έχει τονιστεί προηγούμενα, οι μεταβλητές $\vec{y} \subseteq \mathsf{var}(\mathcal{Q}_l)$ δεν επηρεάζονται από αυτό τον συμπερασμό και συνεπώς θα πρέπει να έχουμε $\vec{y} \subseteq \mathsf{var}(\mathcal{Q}_0)$. Άρα, η επέκταση του $\mathcal{Q}_0$ με το $\vec{y}$ είναι ασφαλής και προφανώς η επέκταση του είναι το ερώτημα $\mathcal{Q}'_e$. Ακόμη, εφόσον η $\mathcal{R}$ είναι κλειστή ως προς το συμπερασμό έχουμε επίσης ότι $\mathcal{Q}_0 \in \mathcal{R}$. Συνεπώς, υπάρχει ένα ερώτημα $\mathcal{Q}_0$ στην $\mathcal{R}$ τ.ω. η επέκταση του με το $\vec{y}$ είναι ασφαλής και υπάγει το $\mathcal{Q}'_{l+1}$.

Από τα παραπάνω προκύπτει η απόδειξη της Ιδιότητας (VE).Έστω τώρα ότι όταν εφαρμόζεται στα $\mathcal{Q}', \mathcal{T}$, ο αλγόριθμος rew τερματίζει ύστερα από n βήματα έχοντας παράγει την $\mathcal{R}_n$. Η Ιδιότητα (VE) υποδηλώνει ότι κατά τον τερματισμό του Αλγορίθμου 4 για κάθε ερώτημα $\mathcal{Q}_l \in \mathcal{R}_n$ θα υπάρχει ένα ερώτημα $\mathcal{Q}'$ στην $\mathcal{R}'$ το οποίο θα υπάγει το $\mathcal{Q}_l$.

Τέλος, είναι προφανές ότι η έξοδος του αλγορίθμου θα είναι επίσης κλειστή προς το συμπερασμό. □

3.3 Μείωση Ερωτημάτων με Άτομα

Ακολούθως, στην ενότητα αυτή μελετάμε το πρόβλημα υπολογισμού μιας επανεγγραφής για ένα ερώτημα του οποίου έχουν αφαιρεθεί κάποια άτομα από το σώμα του, χρησιμοποιώντας και πάλι όσο το δυνατόν περισσότερη πληροφορία από την επενεγγραφή που έχει υπολογιστεί προηγούμενα για το αρχικό ερώτημα. Το παρακάτω παράδειγμα περιγράφει τις βασικές ιδέες στις οποίες στηρίζεται ο αλγόριθμος που θα παρουσιάσουμε στη συνέχεια.

Παράδειγμα 3.3.1. Έστω το TBox $\mathcal{T}$ που δίνεται στη συνέχεια σε μορφή προτάσεων:

$$\mathsf{ΜεταπτυχιακόςΦοιτητής}(x) \rightarrow \mathsf{Φοιτητής}(x), \tag{3.4}$$

$$\mathsf{Τενίστας}(x) \rightarrow \mathsf{Αθλητής}(x) \tag{3.5}$$

και έστω επίσης το ερώτημα

$$\mathcal{Q}_1 = \mathsf{Φοιτητής}(x_1) \wedge \mathsf{Αθλητής}(x_1) \rightarrow \mathcal{Q}_A(x_1).$$

Το σύνολο $\mathcal{R} = \{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3, \mathcal{Q}_4\}$, όπου τα $\mathcal{Q}_i$ ορίζονται στη συνέχεια, αποτελεί επανεγγραφή για το $\mathcal{Q}_1$ μ.β.τ. $\mathcal{T}$ και μπορεί να υπολογιστεί χρησιμοποιώντας συστήματα όπως το PerfectRef, το Rapid, και το Requiem:

$$\begin{aligned}
\mathcal{Q}_2 &= \textsf{Φοιτητής}(x_2) \wedge \textsf{Τενίστας}(x_2) \rightarrow \mathcal{Q}_A(x_2),\\
\mathcal{Q}_3 &= \textsf{ΜεταπτυχιακόςΦοιτητής}(x_3) \wedge \textsf{Αθλητής}(x_3) \rightarrow \mathcal{Q}_A(x_3),\\
\mathcal{Q}_4 &= \textsf{ΜεταπτυχιακόςΦοιτητής}(x_4) \wedge \textsf{Τενίστας}(x_4) \rightarrow \mathcal{Q}_A(x_4)
\end{aligned}$$

Επιπρόσθετα, $\rightsquigarrow_{\mathcal{R}} = \{\langle x_1, x_2\rangle, \langle x_1, x_3\rangle, \langle x_1, x_4\rangle\}$, η οποία δείχνει πως σχετίζεται η μεταβλητή x_1 με τις μεταβλητές x_2, x_3, και x_4.

Πιο συγκεκριμένα, έχουμε τους παρακάτω συμπερασμούς και τις αντίστοιχες υποσημειώσεις $\Upsilon_i = \{\langle \mathcal{A}_i, \mathcal{B}_i\rangle\}$:

$$\begin{aligned}
&\langle \mathcal{Q}_1, (3.5), \mathcal{Q}_2\rangle \quad \text{με} \quad \mathcal{A}_2 = \{\textsf{Αθλητής}(x_1)\} \text{ και } \mathcal{B}_2 = \{\textsf{Τενίστας}(x_2)\}\\
&\langle \mathcal{Q}_1, (3.4), \mathcal{Q}_3\rangle \quad \text{με} \quad \mathcal{A}_3 = \{\textsf{Φοιτητής}(x_1)\} \text{ και } \mathcal{B}_3 = \{\textsf{ΜεταπτυχιακόςΦοιτητής}(x_3)\}\\
&\langle \mathcal{Q}_2, (3.4), \mathcal{Q}_4\rangle \quad \text{με} \quad \mathcal{A}_4 = \{\textsf{Φοιτητής}(x_2)\} \text{ και } \mathcal{B}_4 = \{\textsf{ΜεταπτυχιακόςΦοιτητής}(x_4)\}\\
&\langle \mathcal{Q}_3, (3.5), \mathcal{Q}_4\rangle \quad \text{με} \quad \mathcal{A}_5 = \{\textsf{Αθλητής}(x_3)\} \text{ και } \mathcal{B}_5 = \{\textsf{Τενίστας}(x_4)\}
\end{aligned}$$

Έστω τώρα ότι στη συνέχεια ο χρήστης θέλει να ανακτήσει όλα τα αντικείμενα που είναι φοιτητές—δηλαδή, να θέσει το ερώτημα

$$\mathcal{Q}'_1 = \textsf{Φοιτητής}(x'_1) \rightarrow \mathcal{Q}_A(x'_1).$$

Εύκολα μπορεί να δειχθεί ότι $\mathcal{R}' = \{\mathcal{Q}'_1, \mathcal{Q}'_2\}$, με $\mathcal{Q}'_2 = \textsf{ΜεταπτυχιακόςΦοιτητής}(x'_2) \rightarrow \mathcal{Q}_A(x'_2)$, είναι επανεγγραφή του $\mathcal{Q}'_1$ μ.β.τ. $\mathcal{T}$ που μπορεί να υπολογιστεί από οποιοδήποτε αλγόριθμο/σύστημα.

Παρόλα αυτά, παρατηρούμε ότι μια επανεγγραφή για το $\mathcal{Q}'_1$ μ.β.τ. $\mathcal{T}$ μπορεί να υπολογιστεί χρησιμοποιώντας την πληροφορία που έχει υπολογιστεί προηγούμενα για το $\mathcal{R}$. Πιο συγκεκριμένα, ένα ερώτημα ισοδύναμο με το $\mathcal{Q}'_1$ μπορεί να παραχθεί αφαιρώντας το άτομο $\textsf{Αθλητής}(x_1)$ από το ερώτημα $\mathcal{Q}_1$, ενώ ένα ερώτημα ισοδύναμο με το $\mathcal{Q}'_2$ μπορεί να παραχθεί αφαιρώντας από το $\mathcal{Q}_3$ το άτομο $\textsf{Αθλητής}(x_1)\tau_3$, όπου η αντικατάσταση $\tau_3 = \{x_1 \mapsto x_3\}$ ορίζεται με βάση τη σχέση $\rightsquigarrow_{\mathcal{R}}$. Τέλος, παρατηρούμε ότι τα ερωτήματα $\mathcal{Q}_2$ και $\mathcal{Q}_4$ πρέπει να απορριφθούν καθώς για κάθε αντικατάσταση σ έχουμε ότι $\textsf{Αθλητής}(x_1)\sigma \notin \textsf{bd}(\mathcal{Q}_2)$ και $\textsf{Αθλητής}(x_1)\sigma \notin \textsf{bd}(\mathcal{Q}_4)$. Διαισθητικά, αυτό οφείλεται στο γεγονός ότι το άτομο $\textsf{Αθλητής}(x_1)$ που αφαιρείται «συμμετέχει» στο συμπερασμό $\langle \mathcal{Q}_1, (3.5), \mathcal{Q}_2\rangle$, δηλαδή, $\textsf{Αθλητής}(x_1) \in \mathcal{A}_2$, και στο συμπερασμό $\langle \mathcal{Q}_3, (3.5), \mathcal{Q}_4\rangle$, δηλαδή, $\textsf{Αθλητής}(x_1)\tau_3 \in \mathcal{A}_5$, όπου $\tau_3 = \{x_1 \mapsto x_3\}$. Άρα, στο μειωμένο ερώτημα οι συμπερασμοί αυτοί δεν είναι δυνατοί. Παρατηρούμε ακόμα ότι το $\mathcal{Q}_4$ παράγεται και από το συμπερασμό $\langle \mathcal{Q}_2, (3.4), \mathcal{Q}_4\rangle$, αλλά εφόσον το $\mathcal{Q}_2$ αφαιρείται ο συμπερασμός αυτός δεν είναι εφικτός. $\diamondsuit$

Το παραπάνω παράδειγμα δείχνει ότι δεδομένης μιας επανεγγραφής $\mathcal{R}$ για ένα ερώτημα $\mathcal{Q}$, μια βασική λειτουργία για τον υπολογισμό μιας επανεγγραφής για ένα ερώτημα $\mathcal{Q}'$ για το οποίο ισχύει $\textsf{bd}(\mathcal{Q}') = \textsf{bd}(\mathcal{Q}) \setminus \{\alpha\}$ είναι η «αφαίρεση» του ατόμου $\alpha\tau$ από κάθε ΣΕ $\mathcal{Q}_i \in \mathcal{R}$ τ.ω. $\alpha\tau \in \textsf{bd}(\mathcal{Q}_i)$, όπου η αντικατάσταση τ προκύπτει από

τη σχέση $\rightsquigarrow_{\mathcal{R}}$. Διαφορετικά το $\mathcal{Q}_i$ θα πρέπει να απορρίπτεται. Επιπλέον, δείχνει ότι ο αλγόριθμος μπορεί να χρησιμοποιήσει επιπλέον πληροφορία που παράγεται κατά την κατασκευή της επανεγγραφής $\mathcal{R}$, όπως είναι τα σύνολα $\mathcal{A}_i$, έτσι ώστε να μπορεί να αποφανθεί εάν ένας συμπερασμός είναι εφικτός ακόμα και με την αφαίρεση κάποιων ατόμων.

Το παρακάτω παράδειγμα καταδεικνύει ένα δεύτερο τεχνικό σημείο που μπορεί να προκύψει όταν υπολογίζουμε την επανεγγραφή ενός ερωτήματος που έχει μειωθεί, που και πάλι χρειάζεται επιπλέον πληροφορία από την είσοδο.

Παράδειγμα 3.3.2. Έστω το παρακάτω TBox και ερώτημα:

$$\begin{aligned}\mathcal{T} &= \{\textsf{Φοιτητής}(x) \rightarrow \exists y.\textsf{παρακολουθεί}(x,y),\\ &\quad\ \textsf{ΜεταπτυχιακόςΦοιτητής}(x) \rightarrow \textsf{Φοιτητής}(x)\}\\ \mathcal{Q}_1 &= \textsf{Φοιτητής}(x_1) \wedge \textsf{παρακολουθεί}(x_1,y_1) \rightarrow \mathcal{Q}_A(x_1)\end{aligned}$$

Μια επανεγγραφή για το $\mathcal{Q}_1$ μ.β.τ. $\mathcal{T}$ αποτελείται από το σύνολο $\mathcal{R} = \{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3\}$, όπου $\mathcal{Q}_2 = \textsf{Φοιτητής}(x_2) \rightarrow \mathcal{Q}_A(x_2)$ και $\mathcal{Q}_3 = \textsf{ΜεταπτυχιακόςΦοιτητής}(x_3) \rightarrow \mathcal{Q}_A(x_3)$. Πιο συγκεκριμένα, τα αξιώματα του $\mathcal{T}$ μετασχηματίζονται στις παρακάτω προτάσεις

$$\textsf{Φοιτητής}(x) \rightarrow \textsf{παρακολουθεί}(x, f(x)), \tag{3.6}$$

$$\textsf{ΜεταπτυχιακόςΦοιτητής}(x) \rightarrow \textsf{Φοιτητής}(x) \tag{3.7}$$

και το $\mathcal{Q}_2$ παράγεται από ένα συμπερασμό με κύρια υπόθεση το $\mathcal{Q}_1$, παράπλευρη πρόταση την (3.6) και $\Upsilon_2 = \{\langle \mathcal{A}_2, \mathcal{B}_2\rangle\}$ όπου $\mathcal{A}_2 = \{\textsf{παρακολουθεί}(x_1,y_1)\}$ και $\mathcal{B}_2 = \{\textsf{Φοιτητής}(x_2)\}$, ενώ το $\mathcal{Q}_3$ παράγεται από ένα συμπερασμό με το $\mathcal{Q}_2$ ως κύρια υπόθεση, την (3.7) ως παράπλευρη υπόθεση και $\Upsilon_3 = \{\langle \mathcal{A}_3, \mathcal{B}_3\rangle\}$, όπου $\mathcal{A}_3 = \{\textsf{Φοιτητής}(x_2)\}$ και $\mathcal{B}_3 = \{\textsf{ΜεταπτυχιακόςΦοιτητής}(x_3)\}$. Επιπλέον, έχουμε $\rightsquigarrow_{\mathcal{R}} = \{\langle x_1, x_2\rangle, \langle x_1, x_3\rangle\}$.

Έστω τώρα ότι θέλουμε να αφαιρέσουμε το άτομο $\alpha = \textsf{Φοιτητής}(x_1)$ και συνεπώς να υπολογίσουμε μια επανεγγραφή για το ερώτημα

$$\mathcal{Q}'_1 = \textsf{παρακολουθεί}(x'_1, y'_1) \rightarrow \mathcal{Q}_A(x'_1)$$

όπως είναι για παράδειγμα η επανεγγραφή $\mathcal{R}' = \{\mathcal{Q}'_1, \mathcal{Q}_2, \mathcal{Q}_3\}$.

Έστω ότι θέλουμε να υπολογίσουμε την επανεγγραφή $\mathcal{R}'$ χρησιμοποιώντας την προσέγγιση μας. Πρώτα, αφαιρούμε το άτομο $\textsf{Φοιτητής}(x_1)$ από το $\mathcal{Q}_1$ και συνεπώς παίρνουμε το $\mathcal{Q}'_1$. Στη συνέχεια, για $\tau = \{x_1 \mapsto x_2\}$ μπορούμε να αφαιρέσουμε το $\textsf{Φοιτητής}(x_1)\tau$ από το $\mathcal{Q}_2$, όμως, κάτι τέτοιο έχει ως αποτέλεσμα την παραγωγή μιας πρότασης που δεν είναι έγγυρη. Ο λόγος είναι ότι το άτομο $\textsf{Φοιτητής}(x_1)\tau$ εμφανίζεται στο $\mathcal{Q}_2$ επειδή έχει προστεθεί από ένα συμπερασμό με το $\mathcal{Q}_1$ ως κύρια υπόθεση πάνω στο άτομο του $\mathcal{Q}_1$ που είναι «διαφορετικό» από αυτό που αφαιρείται, δηλαδή, $\mathcal{A}_2 = \{\textsf{παρακολουθεί}(x_1,y_1)\} \not\ni \textsf{Φοιτητής}(x_1)\tau$ και $\textsf{Φοιτητής}(x_1)\tau \in \mathcal{B}_2$. Διαισθητικά, αυτό συνεπάγεται ότι η κατασκευή αυτού του κομματιού του ερωτήματος από το σύστημα συμπερασμού είναι ανεξάρτητη από την αφαίρεση του ατόμου α. Συνεπώς, ο αλγόριθμος μας θα πρέπει να «αντιγράψει» τα αντίστοιχα ερωτήματα, δηλαδή, τα $\mathcal{Q}_2$ και $\mathcal{Q}_3$ (το $\mathcal{Q}_3$ παράγεται από το $\mathcal{Q}_2$) στο αποτέλεσμα, παράγοντας την επιθυμητή επανεγγραφή. $\diamondsuit$

Μια παρόμοια κατάσταση προκύπτει όταν έχουμε ένα συμπερασμό της μορφής $\langle \mathcal{Q}_1, C, \mathcal{Q}_2 \rangle$ όπου $\mathcal{Q}_1 = R(x,y) \wedge R(z,y) \rightarrow \mathcal{Q}_A(x)$, $C = A(x) \rightarrow R(x, f(x))$, $\mathcal{Q}_2 = A(x) \rightarrow \mathcal{Q}_A(x)$ και $\mathcal{A}_1 = \{R(x,y), R(z,y)\}$, ενώ το άτομο που αφαιρείται είναι το $R(z,y)$. Παρόλο που $R(z,y) \notin \mathsf{bd}(\mathcal{Q}_2)$ και $R(z,y) \in \mathcal{A}_1$, δηλαδή, το άτομο που αφαιρείται συμμετέχει στο συμπερασμό που παράγει το ερώτημα $\mathcal{Q}_2$, ο συμπερασμός είναι δυνατός ακόμα και μετά την αφαίρεση του ατόμου, και συνεπώς ο αλγόριθμος θα πρέπει και πάλι να αντιγράψει το $A(x) \rightarrow \mathcal{Q}_A(x)$ στο αποτέλεσμα. Τόσο στην περίπτωση αυτή όπως και στο Παράδειγμα 3.3.2 η διαίσθηση είναι ότι υπάρχουν ερωτήματα που είναι «ανεξάρτητα» από την αφαίρεση κάποιου ατόμου και συνεπώς αυτά θα πρέπει να να αντιγράφονται στο αποτέλεσμα. Στη συνέχεια ορίζουμε τυπικά αυτή την ιδιότητα.

Ορισμός 3.3.3. Έστω $\mathcal{Q}$ ΣΕ, έστω $\mathcal{T}$ TBox, και έστω $\mathcal{R} = \mathcal{R}_D \uplus \mathcal{R}_Q$ επανεγγραφή για το $\mathcal{Q}$ μ.β.τ. $\mathcal{T}$ που προκύπτει από ένα αλγόριθμο επανεγγραφής χρησιμοποιώντας ένα σύστημα συμπερασμού Γ. Ένα ερώτημα $\mathcal{Q}' \in \mathcal{R}_Q$ θα λέγεται *ανεξάρτητο* (*independent*) από το α εάν ισχύει μια από τις παρακάτω συνθήκες:

- το $\mathcal{Q}'$ είναι το αποτέλεσμα ενός συμπερασμού με υπόθεση ένα ερώτημα $\mathcal{Q}'' \in \mathcal{R}_Q$ και υποσημείωση Υ τ.ω. για την αντικατάσταση $\tau'' := \{x \mapsto y \mid x \in \mathsf{var}(\mathcal{Q}), y \in \mathsf{var}(\mathcal{Q}''), x \rightsquigarrow_{\mathcal{R}} y\}$ έχουμε $\alpha\tau'' \in \mathsf{bd}(\mathcal{Q}'')$ και για καποια $\langle \mathcal{A}, \mathcal{B} \rangle \in \Upsilon$ και $\tau' := \{x \mapsto y \mid x \in \mathsf{var}(\mathcal{Q}), y \in \mathsf{var}(\mathcal{Q}'), x \rightsquigarrow_{\mathcal{R}} y\}$ ισχύει $\mathcal{A} \supset \{\alpha\tau''\}$ ή $\alpha\tau' \in \mathcal{B}$, ή
- υπάρχει $\mathcal{Q}_j \in \mathcal{R}_Q$ τ.ω. το $\mathcal{Q}_j$ είναι ανεξάρτητο από το α και το $\mathcal{Q}'$ προκύπτει από το $\mathcal{T} \cup \{\mathcal{Q}_j\}$ στο Γ.

$\triangle$

Στο Παράδειγμα 3.3.2, το ερώτημα $\mathcal{Q}_2$ είναι ανεξάρτητο από το άτομο $\mathsf{Φοιτητής}(x_1)$ καθώς προκύπτει από ένα συμπερασμό με κύρια υπόθεση το $\mathcal{Q}_1$, επίσης $\mathsf{Φοιτητής}(x_1) \in \mathsf{bd}(\mathcal{Q}_1)$ (τ'' είναι η ταυτοτική αντικατάσταση) και για $\tau' = \{x_1 \mapsto x_2\}$ και $\Upsilon_2 = \{\langle \mathcal{A}_2, \mathcal{B}_2 \rangle\}$ (υποσημείωση του συγκεκριμένου συμπερασμού) έχουμε $\mathsf{Φοιτητής}(x_1)\tau' \in \mathcal{B}_2$. Επιπλέον, το $\mathcal{Q}_3$ είναι επίσης ανεξάρτητο από το $\mathsf{Φοιτητής}(x_1)$, εφόσον το $\mathcal{Q}_3$ παράγεται από το $\mathcal{Q}_2$ που όπως έχει δειχθεί προηγουμένως είναι ανεξάρτητο από το $\mathsf{Φοιτητής}(x_1)$.

Όπως δείχνει το επόμενο παράδειγμα, οι πράξεις που περιγράψαμε παραπάνω μπορεί να μην είναι αρκετές για τον υπολογισμό μιας επανεγγραφής για το νέο (μειωμένο) ερώτημα. Επιπρόσθετα, όπως στην περίπτωση της προβολής ερωτήματος, υπάρχουν περιπτώσεις που επιπλέον συμπερασμοί είναι απαραίτητοι για την κατασκευή μιας πλήρους επανεγγραφής για τα μειωμένα ερωτήματα.

Παράδειγμα 3.3.4. Έστω το παρακάτω TBox και ερώτημα:

$$\mathcal{T} = \{A(x) \rightarrow \exists y.R(x,y)\} \qquad \mathcal{Q}_1 = R(x_1, y_1) \wedge B(y_1) \rightarrow \mathcal{Q}_A(x_1)$$

Το σύνολο $\mathcal{R} = \{\mathcal{Q}_1\}$ είναι μια επανεγγραφή για το $\mathcal{Q}_1$ μ.β.τ. $\mathcal{T}$.

Έστω τώρα ότι θέλουμε να υπολογίσουμε μια επανεγγραφή για το ερώτημα $\mathcal{Q}_1' = R(x_1', y_1') \rightarrow \mathcal{Q}_A(x_1')$ χρησιμοποιώντας την προσέγγιση μας. Κάτι τέτοιο θα

είχε ως αποτέλεσμα την παραγωγή του συνόλου $\mathcal{R}' = \{\mathcal{Q}'_1\}$ το οποίο όμως δεν είναι επανεγγραφή για το ερώτημα $\mathcal{Q}'_1$ μ.β.τ. $\mathcal{T}$. Πιο συγκεκριμένα, για το αντικείμενο $I = \{A(a)\}$ έχουμε $\mathsf{cert}(\mathcal{Q}'_1, \mathcal{T} \cup I) = \{a\}$, ενώ $\mathsf{cert}(\mathcal{R}', I) = \emptyset$.

Για να υπολογίσουμε μια επανεγγραφή για το $\mathcal{Q}'_1$ πρέπει αρχικά να μετασχηματίσουμε το μοναδικό αξίωμα του $\mathcal{T}$ στην πρόταση

$$A(x) \rightarrow R(x, f(x)) \tag{3.8}$$

Στη συνέχεια, χρειάζεται να εφαρμόσουμε κανόνες συμπεραμού από ένα αλγόριθμο επανεγγραφής για το ερώτημα $\mathcal{Q}'_1$. Αυτό θα έχει ως αποτέλεσμα την παραγωγή του ερωτήματος $\mathcal{Q}'_2 = A(x'_2) \rightarrow \mathcal{Q}_A(x'_2)$ από το $\mathcal{Q}'_1$ και την πρόταση (3.8). Τότε μπορεί να επαληθευτεί ότι το σύνολο $\mathcal{R}' = \{\mathcal{Q}'_1, \mathcal{Q}'_2\}$ είναι επανεγγραφή για τα $\mathcal{Q}'_1, \mathcal{T}$. $\diamondsuit$

Και πάλι ο αλγόριθμος μας στηρίζεται στην παρακάτω συνάρτηση για να μπορεί να αναγνωρίσει τις περιπτώσεις για τις οποίες είναι απαραίτητη η εφαρμογή επιπλέον κανόνων συμπερασμού.

Ορισμός 3.3.5. Έστω $\mathcal{Q}$ ΣΕ, έστω α άτομο του $\mathcal{Q}$ και έστω $\mathcal{T}$ TBox. Τότε, η συνάρτηση $\mathsf{needsRewriting}(\mathcal{Q}, \alpha, \mathcal{T})$ επιστρέφει **αληθές** εάν υπάρχει συμπερασμός με κύρια υπόθεση το ερώτημα $\bigwedge \mathsf{bd}(\mathcal{Q}) \setminus \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q})$ και υποσημειώσεις $\sigma_\alpha, \Upsilon_\alpha$ και υπάρχει $\langle \mathcal{A}_\alpha, \mathcal{B}_\alpha \rangle \in \Upsilon_\alpha$ τ.ω. για κάθε συμπερασμό με κύρια υπόθεση το $\mathcal{Q}$ και υποσημειώσεις σ, Υ και κάθε $\langle \mathcal{A}, \mathcal{B} \rangle \in \Upsilon$ είτε $\mathcal{A} \neq \mathcal{A}_\alpha$ ή $\mathcal{B} \neq \mathcal{B}_\alpha$. Διαφορετικά επιστρέφει **ψευδές**. $\triangle$

Τέλος, όπως στην περίπτωση της επέκτασης ερωτημάτων με νέες μεταβλητές, το επόμενο παράδειγμα δείχνει ότι η επανεγγραφή που χρησιμοποιείται στην είσοδο του αλγορίθμου πρέπει να είναι κλειστή ως προς το συμπερασμό, διαφορετικά είναι απαραίτητη η εφαρμογή συμπερασμών για όλα τα μειωμένα ερωτήματα.

Παράδειγμα 3.3.6. Έστω τα $\mathcal{T}$ και $\mathcal{Q}$ από το Παράδειγμα 3.2.3 καθώς και η κλειστή ως προς το συμπερασμό επανεγγραφή $\mathcal{R} = \{\mathcal{Q}_1, \mathcal{Q}_2\}$, όπου $\mathcal{Q}_2 = A(x_2) \rightarrow \mathcal{Q}_A(x_2)$ και η επανεγγραφή $\mathcal{R}' = \{\mathcal{Q}_2\}$ που δεν είναι κλειστή ως προς το συμπερασμό.

Έστω τώρα ότι θέλουμε να αφαιρέσουμε το άτομο $\alpha = A(x_1)$, δηλαδή, θέλουμε να υπολογίσουμε την επανεγγραφή για το ερώτημα $\mathcal{Q}'_1 = R(x_1, y_1) \rightarrow \mathcal{Q}_A(x_1)$. Μια τέτοια επανεγγραφή είναι η $\mathcal{R}'' = \{\mathcal{Q}'_1, \mathcal{Q}_2\}$ που μπορεί να υπολογιστεί από όλα τα σύγγχρονα συστήματα επανεγγραφής. Παρόλα αυτά, είναι εύκολο να παρατηρήσουμε ότι χρησιμοποιώντας τη μέθοδο που περιγράψαμε παραπάνω δεν μπορούμε να πάρουμε την $\mathcal{R}''$ από την $\mathcal{R}'$. Μπορεί όμως, να υπολογιστεί χρησιμοποιώντας την $\mathcal{R}$. $\diamondsuit$

Ο Αλγόριθμος 5 παρουσιάζει την προσέγγιση μας για τον υπολογισμό μιας επανεγγραφής για ένα μειωμένο ερώτημα εκμεταλλευόμενοι την πληροφορία που έχει παραχθεί προηγούμενα και στηρίζεται στα σημεία που παρουσιάζουμε στη συνέχεια. Ο αλγόριθμος δέχεται ως είσοδο ένα TBox $\mathcal{T}$, ένα ερώτημα $\mathcal{Q}$, μια επανεγγραφή $\mathcal{R}$ για το $\mathcal{Q}$ μ.β.τ. $\mathcal{T}$ και ένα άτομο $\alpha \in \mathcal{Q}$. Επιπλέον στηρίζει τη λειτουργία του σε ένα οποιοδήποτε αλγόριθμο επανεγγραφής rew έτσι ώστε να μπορεί να παράγει επανεγγραφές όποτε αυτό κρίνεται απαραίτητο.

Ο αλγόριθμος αρχικοποιεί πρώτα μια νέα επανεγγραφή $\mathcal{R}'$ με την $\mathcal{R}_D$ και τη σχέση $\rightsquigarrow_{\mathcal{R}'}$ με την $\rightsquigarrow_{\mathcal{R}}$, περιορισμένη όμως στις μεταβλητές που εμφανίζονται στα

Αλγόριθμος 5 removeAtom$(\mathcal{T}, \mathcal{Q}, \mathcal{R}, \rightsquigarrow_{\mathcal{R}}, \alpha)$

Είσοδος: Ένα TBox $\mathcal{T}$, ένα ερώτημα $\mathcal{Q}$, μια επανεγγραφή $\langle \mathcal{R}, \rightsquigarrow_{\mathcal{R}} \rangle$ για το $\mathcal{Q}$ μ.β.τ. $\mathcal{T}$ με διαμέριση $\mathcal{R}_D \uplus \mathcal{R}_Q$, και ένα άτομο $\alpha \in \mathsf{bd}(\mathcal{Q})$.

1: Αρχικοποίησε μια επανεγγραφή $\mathcal{R}' := \mathcal{R}_D$
2: $\rightsquigarrow_{\mathcal{R}'} := \{\langle x, y \rangle \mid x \rightsquigarrow_{\mathcal{R}} y$ και $x \in \mathsf{var}(\mathsf{bd}(\mathcal{Q}) \setminus \{\alpha\})\}$
3: Έστω ένας τελεστής προβολής π που περιλαμβάνει τις θέσεις k για όλες τις μεταβλητές του $\mathsf{avar}(\mathcal{Q})$
4: **if** $y \in \mathsf{var}(\alpha)$ with $y \in \mathsf{avar}(\mathcal{Q}) \wedge y \notin \mathsf{var}(\mathsf{bd}(Q) \setminus \{\alpha\})$ **then**
5: Αφαίρεσε από το π κάθε θέση j_k για την οποία $x_{j_k} = y$
6: **end if**
7: **for** $\mathcal{Q}_i \in \mathcal{R}_Q$ **do**
8: $\tau := \{x \mapsto y \mid x \in \mathsf{var}(\mathcal{Q}), y \in \mathsf{var}(\mathcal{Q}_i)$ και $x \rightsquigarrow_{\mathcal{R}} y\}$
9: **if** $\mathcal{Q}_i$ ανεξάρτητο από το α **then**
10: Προσέθεσε το $\mathcal{Q}_i$ στο $\mathcal{R}'_Q$
11: **else if** $\alpha\tau \in \mathsf{bd}(\mathcal{Q}_i)$ **then**
12: $\mathcal{Q}'_i := \pi(\bigwedge \mathsf{bd}(\mathcal{Q}_i) \setminus \{\alpha\tau\} \rightarrow \mathsf{hd}(\mathcal{Q}_i))$
13: Προσέθεσε το $\mathcal{Q}'_i$ στο $\mathcal{R}'_Q$
14: **if** needsRewriting$(\mathcal{Q}_i, \alpha\tau, \mathcal{T})$ **then**
15: $\langle \mathcal{R}_n, \rightsquigarrow_{\mathcal{R}_n} \rangle := \mathsf{rew}(\mathcal{Q}'_i, \mathcal{T})$
16: $\mathcal{R}' := \mathcal{R}' \cup \mathcal{R}_n$ και $\rightsquigarrow_{\mathcal{R}'} := \rightsquigarrow_{\mathcal{R}'} \cup \rightsquigarrow_{\mathcal{R}_n}$
17: **end if**
18: **end if**
19: **end for**
20: **return** $\langle \mathcal{R}', \rightsquigarrow_{\mathcal{R}'} \rangle$

άτομα $\mathsf{bd}(\mathcal{Q}) \setminus \{\alpha\}$. Πριν συνεχίσουμε στην περιγραφή του αλγορίθμου, αξίζει να σημειωθεί πως ο αλγόριθμος ελέγχει πρώτα εάν το άτομο που πρόκειται να αφαιρεθεί περιέχει μια διακεκριμένη μεταβλητή του $\mathcal{Q}$ που δεν εμφανίζεται σε κανένα άλλο άτομο του $\mathcal{Q}$ (γραμμή 4). Στην περίπτωση αυτή, εκτός από την αφαίρεση του ατόμου από κάθε ερώτημα, ο αλγόριθμος χρειάζεται να αφαιρέσει επίσης τη μεταβλητή αυτή από την κεφαλή κάθε ερωτήματος. Κάτι τέτοιο είναι εφικτό με την κατασκευή ενός κατάλληλου τελεστή προβολής π.

Στη συνέχεια, ο αλγόριθμος εισέρχεται στον κεντρικό βρόχο. Προσπελαύνει κάθε ερώτημα $\mathcal{Q}_i \in \mathcal{R}_Q$ και παράγει κατάλληλες αντιστοιχήσεις τ των μεταβλητών του $\mathcal{Q}_i$ μ.β.τ. τις μεταβλητές του $\mathcal{Q}$ (γραμμή 8). Στη συνέχεια, ελέγχει εάν το $\mathcal{Q}_i$ είναι ανεξάρτητο από το α (γραμμή 9) και εάν κάτι τέτοιο ισχύει τότε προσθέτει το $\mathcal{Q}_i$ στο αποτέλεσμα. Διαφορετικά εάν το α εμφανίζεται στο σώμα του $\mathcal{Q}_i$ (γραμμή 11) κατασκευάζει το νέο ερώτημα $\mathcal{Q}'_i := \bigwedge \mathsf{bd}(\mathcal{Q}_i) \setminus \{\alpha\tau\} \rightarrow \mathsf{hd}(\mathcal{Q}_i)$ (γραμμή 12) και το προσθέτει στο αποτέλεσμα. Τέλος, χρησιμοποιεί τη συνάρτηση needsRewriting για να ελέγξει εάν το $\mathcal{Q}'_i$ χρειάζεται να επανεγγραφεί περισσότερο και εάν χρειάζεται προσθέτει την παραγόμενη επανεγγραφή στο αποτέλεσμα (γραμμές 14–17).

Θεώρημα 3.3.7. *Έστω $\mathcal{Q}$ ΣΕ, έστω $\mathcal{T}$ TBox, έστω α άτομο τ.ω. $\alpha \in \mathsf{bd}(\mathcal{Q})$ και έστω $\mathcal{R}, \rightsquigarrow_{\mathcal{R}}$ έξοδος ενός αλγόριθμου επανεγγραφής όταν εφαρμόζεται στα $\mathcal{Q}, \mathcal{T}$*

έτσι ώστε η $\mathcal{R}$ να είναι κλειστή ως προς το συμπερασμό. Έστω $\mathcal{R}'$ επανεγγραφή που παράγεται από τον Αλγόριθμο 5 όταν εφαρμόζεται στα $\mathcal{T}, \mathcal{Q}, \mathcal{R}, \rightsquigarrow_{\mathcal{R}}$, και α. Τότε το σύνολο $\mathcal{R}'$ είναι μια επανεγγραφή κλειστή ως προς το συμπερασμό για το ερώτημα $\bigwedge \mathsf{bd}(\mathcal{Q}) \setminus \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q})$ μ.β.τ. $\mathcal{T}$.

Απόδειξη: Ο τερματισμός του αλγορίθμου προκύπτει από το γεγονός ότι ο Αλγόριθμος 5 προσπελαύνει μόνο μια φορά κάθε ερώτημα που είναι μέλος ενός πεπερασμένου συνόλου ($\mathcal{R}_Q$) καθώς και επειδή ο αλγόριθμος επανεγγραφής rew τερματίζει.

Έστω, $\mathcal{T}$, $\langle \mathcal{R}, \rightsquigarrow_{\mathcal{R}} \rangle$, $\mathcal{Q}$, και α είσοδος για τον Αλγόριθμο 5, όπου $\langle \mathcal{R}, \rightsquigarrow_{\mathcal{R}} \rangle$ είναι επανεγγραφη κλειστή ως προς το συμπερασμό που υπολογίζεται από κάποιο αλγόριθμο rew και έστω $\alpha \in \mathsf{bd}(\mathcal{Q})$.

Έστω ότι η έξοδος του Αλγόριθμου 5 είναι ένα ζευγάρι $\langle \mathcal{R}', \rightsquigarrow_{\mathcal{R}'} \rangle$ και έστω επίσης ότι $\langle \mathcal{R}_i, \rightsquigarrow_{\mathcal{R}_i} \rangle$ είναι η επανεγγρφαή που υπολογίζεται ύστερα από i βήματα του αλγορίθμου επανεγγραφής rew όταν αυτός εφαρμόζεται για τα $\mathcal{Q}' = \bigwedge \mathsf{bd}(\mathcal{Q}) \setminus \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q})$ και $\mathcal{T}$.

Η απόδειξη του παραπάνω θεωρήματος προκύπτει από την παρακάτω ιδιότητα, την οποία θα δείξουμε χρησιμοποιώντας επαγωγή πάνω στον αριθμό βημάτων i. Για να απλοποιήσουμε τον συμβολισμό και χωρίς βλάβη της γενικότητας υποθέτουμε ότι ο τελεστής προβολής δεν χρειάζεται.

> (AR): Για κάθε $i \geq 0$ και $\mathcal{Q}'_l \in \mathcal{R}_i$ υπάρχει $\mathcal{Q}_l \in \mathcal{R}$ τέτοιο ώστε ισχύει μια από τις παρακάτω συνθήκες:
>
> 1. το $\mathcal{Q}_l$ είναι ανεξάρτητο από το α και υπάγει το $\mathcal{Q}'_l$, ή
> 2. για την αντικατάσταση τ που ορίζεται από τη γραμμή 8 του Αλγορίθμου 5 έχουμε $\alpha\tau \in \mathsf{bd}(\mathcal{Q}_l)$ και για $\mathcal{Q}^r_l = \bigwedge \mathsf{bd}(\mathcal{Q}_l) \setminus \{\alpha\tau\} \rightarrow \mathsf{hd}(\mathcal{Q}_l)$ ισχύει ένα από τα παρακάτω:
> (α′) το $\mathcal{Q}^r_l$ υπάγει το $\mathcal{Q}'_l$, ή
> (β′) η needsRewriting$(\mathcal{Q}_l, \alpha, \mathcal{T})$ επιστρέφει αληθές και κάποιο $\mathcal{Q}''_l \in$ rew$(\mathcal{Q}^r_l, \mathcal{T})$ υπάγει το $\mathcal{Q}'_l$.

Βασική Περίπτωση ($i = 0$): Αρχικά $\mathcal{R}_0 = \{\mathcal{Q}'\}$ όπου $\mathcal{Q}' = \bigwedge \mathsf{bd}(\mathcal{Q}) \setminus \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q})$, συνεπώς, η Ιδιότητα (AR) ικανοποιείται.

Βήμα Επαγωγής: Έστω ότι η Ιδιότητα (AR) ισχύει ύστερα από i βήματα για κάθε $\mathcal{Q}'_l \in \mathcal{R}_i$—δηλαδή, ισχύει είτε το 1., είτε το 2(a), είτε το 2(b). Στη συνέχεια, έστω ότι στο βήμα $i + 1$ ο αλγόριθμος rew εφαρμόζει ένα βήμα συμπερασμού σε κάποιο ερώτημα $\mathcal{Q}'_l$ παράγοντας το $\mathcal{Q}'_{l+1}$. Στη συνέχεια εξετάζουμε κάθε περίπτωση του βήματος επαγωγής ξεχωριστά:

1. το $\mathcal{Q}_l$ είναι ανεξάρτητο από το α και υπάγει το $\mathcal{Q}'_l$. Από τις ιδιότητες της υπαγωγής αυτό συνεπάγεται ότι είτε το $\mathcal{Q}_l$ υπάγει το $\mathcal{Q}'_{l+1}$ ή ότι μπορεί να εφαρμοστεί ένας συμπερασμός στο $\mathcal{Q}_l$ και το αποτέλεσμα, που θα καλούμε $\mathcal{Q}_0$, υπάγει το $\mathcal{Q}'_{l+1}$. Εάν ισχύει η πρώτη περίπτωση τότε η ιδιότητα ικανοποιείται. Εάν ισχύει η δεύτερη περίπτωση τότε και πάλι ισχύει η ιδιότητα καθώς θα έχουμε ότι $\mathcal{Q}_0 \in \mathcal{R}$ και επιπλέον, λόγω του Ορισμού 3.3.3 το $\mathcal{Q}_0$ θα είναι επίσης

ανεξάρτητο από το α (παράγεται από ένα ερώτημα που είναι ανεξάρτητο από το α).

2. $\alpha\tau \in \mathsf{bd}(\mathcal{Q}_l)$, όπου $\tau := \{x \mapsto y \mid x \in \mathsf{var}(\mathcal{Q}), y \in \mathsf{var}(\mathcal{Q}_l)$ και $x \rightsquigarrow_{\mathcal{R}} y\}$ και είτε το $\mathcal{Q}_l^r$ υπάγει το $\mathcal{Q}_l'$ ή η $\mathsf{needsRewriting}(\mathcal{Q}_l, \alpha, \mathcal{T})$ επιστρέφει αληθές και υπάρχει ερώτημα $\mathcal{Q}_l'' \in \mathsf{rew}(\mathcal{Q}_l^r, \mathcal{T})$ που υπάγει το $\mathcal{Q}_l'$. Εξετάζουμε τις δύο αυτές περιπτώσεις ξεχωριστά:

   (α′) το $\mathcal{Q}_l^r$ υπάγει το $\mathcal{Q}_l'$. Αυτό συνεπάγεται είτε ότι το $\mathcal{Q}_l^r$ υπάγει το $\mathcal{Q}_{l+1}'$ ή πως μπορεί να εφαρμοστεί ένας συμπερασμός στο $\mathcal{Q}_l^r$ και το αποτέλεσμα $\mathcal{Q}_0^r$ να υπάγει το $\mathcal{Q}_{l+1}'$. Στην πρώτη περίπτωση η ιδιότητα ισχύει. Έστω τώρα ότι ισχύει η δεύτερη περίπτωση. Υπενθυμίζουμε ότι το $\mathcal{Q}_l$ περιέχει τα ίδια άτομα σώματος με το $\mathcal{Q}_l^r$ εκτός από το $\alpha\tau$. Εφόσον οι αλγόριθμοι επανεγγραφής εφαρμόζουν συμπερασμούς χρησιμοποιώντας μόνο άτομα από το σώμα ενός ερωτήματος, έχουμε ότι όλοι οι συμπερασμοί που εφαρμόζονται στο $\mathcal{Q}_l^r$ μπορούν επίσης να εφαρμοστούν και στο $\mathcal{Q}_l$ παράγοντας το $\mathcal{Q}_{l+1}$. Για τους συμπερασμούς αυτούς το $\bigwedge \mathsf{bd}(\mathcal{Q}_{l+1}) \setminus \{\alpha\tau''\} \rightarrow \mathsf{hd}(\mathcal{Q}_{l+1})$ υπάγει το $\mathcal{Q}_{l+1}'$, όπου $\tau'' = \{x \mapsto y \mid x \in \mathsf{var}(\mathcal{Q}), y \in \mathsf{var}(\mathcal{Q}_{l+1})$ και $x \rightsquigarrow_{\mathcal{R}} y\}$. Παρόλα αυτά, στο $\mathcal{Q}_l$ ένας συμπερασμός μπορεί να εφαρμοστεί και στο $\alpha\tau$. Τέλος, στην περίπτωση που ένας συμπερασμός μπορεί να ερφαμοστεί σε ένα άτομο διαφορετικό από το $\alpha\tau$ που δεν μπορεί να εφαρμοστεί στο $\mathcal{Q}_l$ η συνάρτηση $\mathsf{needsRewriting}(\mathcal{Q}_l, \alpha, \mathcal{T})$ επιστρέφει αληθές και τότε έχουμε $\mathcal{Q}_0^r \in \mathsf{rew}(\mathcal{Q}_l^r, \mathcal{T})$. Συνεπώς, η Ιδιότητα (AR) ισχύει.

   (β′) Παρόμοια με την περίπτωση 2. από την απόδειξη του Θεωρήματος 3.1.3.

Όλα αυτά δείχνουν την Ιδιότητα (AR). Έστω ότι όταν ένας αλγόριθμος επανεγγραφής rew εφαρμόζεται στα $\mathcal{Q}', \mathcal{T}$, τερματίζει ύστερα από n βήματα παράγοντας την επανεγγραφή $\mathcal{R}_n$. Η Ιδιότητα (AR) συνεπάγεται ότι κατά τον τερματισμό του Αλγορίθμου 4 για κάθε ερώτημα $\mathcal{Q}_l \in \mathcal{R}_n$ θα υπάρχει ένα ερώτημα $\mathcal{Q}'$ στο $\mathcal{R}'$ το οποίο θα υπάγει το $\mathcal{Q}_l$. □

Ένα σημείο του Αλγορίθμου 5 το οποίο είναι υπολογιστικά ακριβό είναι ο έλεγχος εάν ένα ερώτημα είναι ανεξάρτητο από το άτομο α (γραμμή 9), όπου γίνεται έλεγχος των συνθηκών του Ορισμού 3.3.3. Όπως φαίνεται και από τον ορισμό κάτι τέτοιο περιλαμβάνει τον έλεγχο διαφόρων σχέσεων παραγωγής ανάμεσα στα μέλη του συνόλου $\mathcal{R}_Q$. Για να μπορέσουμε να υλοποιήσουμε το βήμα αυτό αποδοτικά υποθέτουμε ότι ο αλγόριθμος επανεγγραφής που χρησιμοποιείται για να παράγει την επανεγγραφή $\mathcal{R}$ παρέχει επίσης και την επιπλέον πληροφορία που παράγεται κατά την κατασκευή της $\mathcal{R}$. Πιο συγκεκριμένα, εκτός από την επανεγγραφή $\mathcal{R}$ και τη σχέση $\rightsquigarrow_{\mathcal{R}}$ υποθέτουμε ότι ο αλγόριθμος επανεγγραφής επιστρέφει επίσης μια σχέση $\mathcal{H}$ και μια αντιστοίχιση $\mathcal{S}$ τ.ω. εάν το $\mathcal{Q}'$ είναι το συμπέρασμα ενός συμπερασμού με κύρια υπόθεση το $\mathcal{Q}$ και υποσημείωση Υ, τότε ισχύει $\langle \mathcal{Q}, \mathcal{Q}' \rangle \in \mathcal{H}$ και $\mathcal{S}(\langle \mathcal{Q}, \mathcal{Q}' \rangle) = \Upsilon$. Ο αλγόριθμος μας μπορεί να εκμεταλλευτεί την επιπλέον αυτή πληροφορία για να ελέγξει εάν ένα ερώτημα $\mathcal{Q}_i$ μπορεί να παραχθεί από το $\mathcal{T} \cup \{\mathcal{Q}_j\}$, ελέγχοντας για παράδειγμα εάν το $\mathcal{Q}_i$ είναι προσβάσιμο από το $\mathcal{Q}_j$ στη σχέση $\mathcal{H}$. Επιπλέον, μπορεί να χρησιμοποιήσει τη σχέση $\mathcal{S}$ για να έχει πρόσβαση σε διάφορες υποσημειώσεις χωρίς να είναι απαραίτητο

να εφαρμόσει συμπερασμούς. Όσον αφορά στη χρήση της υπο-ρουτίνας rew και εδώ ισχύουν όσα έχουν συζητηθεί στην ενότητα 3.1.

3.4 Επέκταση Ερωτημάτων με Άτομα

Τέλος, στην ενότητα αυτή μελετάμε το πρόβλημα υπολογισμού μιας επανεγγραφής για ένα ερώτημα το οποίο έχει επεκταθεί με νέα άτομα, χρησιμοποιώντας και πάλι όσο το δυνατόν περισσότερη πληροφορία από την επεναγγραφή που έχει υπολογιστεί προηγούμενα για το αρχικό ερώτημα. Το παρακάτω παράδειγμα περιγράφει τις βασικές ιδέες στις οποίες στηρίζεται ο αλγόριθμος που θα παρουσιάσουμε στη συνέχεια.

Παράδειγμα 3.4.1. Έστω το TBox $\mathcal{T}$ που δίνεται στη συνέχεια:

$$\begin{aligned} \textsf{Καθηγητής}(x) &\rightarrow \exists y.\textsf{διδάσκει}(x,y), \\ \exists y.\textsf{διδάσκει}(y,x) &\rightarrow \textsf{Φοιτητής}(x) \end{aligned}$$

και το ΣΕ

$$\mathcal{Q}_1 = \textsf{διδάσκει}(x_1, y_1) \rightarrow \mathcal{Q}_A(x_1)$$

που ανακτά όλα τα άτομα που διδάσκουν κάποιον. Το σύνολο $\mathcal{R} = \{\mathcal{Q}_1, \mathcal{Q}_2\}$, με $\mathcal{Q}_2 = \textsf{Καθηγητής}(x_2) \rightarrow \mathcal{Q}_A(x_2)$ αποτελεί μια επανεγγραφή για τα $\mathcal{Q}_1, \mathcal{T}$ και μπορεί να υπολογιστεί με ένα οποιοδήποτε αλγόριθμο επανεγγραφής. Επιπρόσθετα η σχέση $\rightsquigarrow_{\mathcal{R}} = \{\langle x_1, x_2\rangle\}$ δείχνει πως σχετίζεται η μεταβλητή x_1 με την x_2. Πιο συγκεκριμένα, τα αξιώματα του $\mathcal{T}$ μετασχηματίζονται στις παρακάτω προτάσεις

$$\textsf{Καθηγητής}(x) \rightarrow \textsf{διδάσκει}(x, f(x)), \tag{3.9}$$

$$\textsf{διδάσκει}(y, x) \rightarrow \textsf{Φοιτητής}(x) \tag{3.10}$$

και το ΣΕ $\mathcal{Q}_2$ παράγεται εφαρμόζοντας ένα συμπερασμό με υποθέσεις το ερώτημα $\mathcal{Q}_1$ και την πρόταση (3.9).

Έστω τώρα ότι στη συνέχεια ο χρήστης θέλει να ανακτήσει όλα τα άτομα που διδάσκουν φοιτητές, δηλαδή, το ερώτημα $\mathcal{Q}_1$ επεκτείνεται με το άτομο $\alpha = \textsf{Φοιτητής}(y_1')$ και το νέο ΣΕ είναι το ακόλουθο:

$$\mathcal{Q}_1' = \textsf{διδάσκει}(x_1', y_1') \wedge \textsf{Φοιτητής}(y_1') \rightarrow \mathcal{Q}_A(x_1').$$

Και πάλι, χρησιμοποιώντας έναν οποιοδήποτε αλγόριθμο επανεγγραφής μπορούμε να υπολογίσουμε την επανεγγραφή $\mathcal{R}' = \{\mathcal{Q}_1', \mathcal{Q}_2', \mathcal{Q}_1, \mathcal{Q}_2\}$ με $\mathcal{Q}_1$ και $\mathcal{Q}_2$ ίδια με αυτά που έχουν υπολογιστεί προηγουμένως και $\mathcal{Q}_2' = \textsf{διδάσκει}(x_2', y_2') \wedge \textsf{διδάσκει}(z_2', y_2') \rightarrow \mathcal{Q}_A(x_2')$, με $\rightsquigarrow_{\mathcal{R}'} = \{\langle x_1', x_2'\rangle, \langle x_1', x_1\rangle, \langle x_1', x_2\rangle, \langle y_1', y_2'\rangle\}$.

Και σε αυτή την περίπτωση όμως, μπορούμε, αντί να εκτελέσουμε έναν αλγόριθμο επανεγγραφής εξαρχής, να εκμεταλλευτούμε την πληροφορία που έχει παραχθεί προηγουμένως, δηλαδή την επανεγγραφή $\mathcal{R}$ για τα $\mathcal{Q}_1$ και $\mathcal{T}$. Πράγματι, το ερώτημα $\mathcal{Q}_1'$ μπορεί να παραχθεί απ' ευθείας από το $\mathcal{R}$ εάν στο σώμα του ερωτήματος $\mathcal{Q}_1$ προστεθεί

το άτομο Φοιτητής(y_1'). Παρατηρούμε όμως, ότι το ερώτημα $\mathcal{Q}_2$ δεν μπορεί να επεκταθεί με το άτομο α μιας και το Καθηγητής$(x_2'') \wedge$ Φοιτητής$(y_2'') \rightarrow \mathcal{Q}_A(x_2'')$ που παράγεται δεν είναι συνδεδεμένο.

Επιπλέον, παρατηρούμε ότι χρησιμοποιώντας αυτή τη μέθοδο δεν μπορούν να παραχθούν τα ερωτήματα $\mathcal{Q}_1'$, $\mathcal{Q}_1$ και $\mathcal{Q}_2$. Για το λόγο αυτό είναι απαραίτητο να χρησιμοποιηθούν οι κανόνες συμπερασμού ενός συστήματος έτσι ώστε να τα υπολογίσουν. Πιο συγκεκριμένα το $\mathcal{Q}_1'$ μπορεί να παραχθεί από ένα κανόνα παραγοντοποίησης με το $\mathcal{Q}_1$ ως κύρια υπόθεση, το $\mathcal{Q}_1$ μπορεί να παραχθεί από ένα συμπερασμό με υποθέσεις το ερώτημα $\mathcal{Q}_1'$ και την πρόταση (3.10), ενώ το $\mathcal{Q}_2$ μπορεί να παραχθεί από ένα συμπερασμό με υποθέσεις το ερώτημα $\mathcal{Q}_1$ και την πρόταση (3.9). $\diamondsuit$

Όλα τα παραπάνω υποδηλώνουν ότι δεδομένου ενός ατόμου α και μιας επανεγγραφής $\mathcal{R}$ για τα $\mathcal{Q}_1, \mathcal{T}$, μια επανεγγραφή για τα $\bigwedge \mathsf{bd}(\mathcal{Q}_1) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q}_1), \mathcal{T}$ μπορεί να υπολογιστεί εάν επεκτείνουμε κατάλληλα κάθε ερώτημα $\mathcal{Q}_i \in \mathcal{R}$ για το οποίο ισχύει $\mathsf{var}(\alpha\tau) \subseteq \mathsf{var}(\mathcal{Q}_i)$ με το άτομο $\alpha\tau$, όπου η αντικατάσταση τ κατασκευάζεται από τη σχέση $\rightsquigarrow_{\mathcal{R}}$, και στη συνέχεια, εφαρμόζοντας (πιθανώς) επιπλέον συμπερασμούς στα ερωτήματα που έχουν προκύψει ώστε να υπολογιστούν αυτά που υπολείπονται. Για να μπορέσουμε να αναγνωρίσουμε περιπτώσεις που είναι απαραίτητη η εφαρμογή επιπλέον κανόνων συμπερασμού χρησιμοποιούμε και πάλι τη συνάρτηση needsRewriting που δίνεται στον Ορισμό 3.3.5 στην οποία όμως τώρα το πρώτο όρισμα αντικαθίσταται από το επεκτεταμένο ερώτημα.

Τέλος, όπως στην περίπτωση της επέκτασης ερωτημάτων με νέες μεταβλητές και της μείωσης ερωτημάτων με άτομα, το επόμενο παράδειγμα δείχνει ότι η επανεγγραφή που χρησιμοποιείται στην είσοδο του αλγορίθμου πρέπει να είναι κλειστή ως προς το συμπερασμό, διαφορετικά είναι απαραίτητη η εφαρμογή συμπερασμών για όλα τα επεκτεταμένα ερωτήματα.

Παράδειγμα 3.4.2. Έστω το παρακάτω TBox και ΣΕ:

$$\mathcal{T} = \{A(x) \rightarrow \exists y.R(x,y)\} \qquad \mathcal{Q}_1 = R(x_1,y_1) \rightarrow \mathcal{Q}_A(x_1)$$

Τότε από την πρόταση $A(x) \rightarrow R(x, f(x))$ που προκύπτει από το αξίωμα του $\mathcal{T}$, και το $\mathcal{Q}$ προκύπτει η επανεγγραφή $\mathcal{R} = \{\mathcal{Q}_1, \mathcal{Q}_2\}$, με το $\mathcal{Q}_2 = A(x_2) \rightarrow \mathcal{Q}_A(x_2)$ να παράγεται από ένα συμπερασμό στο $\mathcal{Q}_1$ και $\rightsquigarrow_{\mathcal{R}} = \{\langle x_1, x_2\rangle\}$. Παρόλα αυτά, παρατηρούμε ότι το ερώτημα $\mathcal{Q}_2$ υπάγει το $\mathcal{Q}_1$, συνεπώς το $\mathcal{Q}_1$ μπορεί να αφαιρεθεί και το σύνολο $\mathcal{R}' = \{\mathcal{Q}_2\}$ είναι επίσης μια επανεγγραφή για τα $\mathcal{Q}_1, \mathcal{T}$ η οποία όμως δεν είναι κλειστή ως προς το συμπερασμό.

Έστω τώρα ότι θέλουμε να επεκτείνουμε το ερώτημα $\mathcal{Q}_1$ με το άτομο $B(y_1)$, θέλουμε δηλαδή να υπολογίσουμε μια επανεγγραφή για το ερώτημα $\mathcal{Q}_1' = B(y_1) \wedge R(x_1,y_1) \rightarrow \mathcal{Q}_A(x_1)$. Μια τέτοια επανεγγραφή είναι η $\mathcal{R}'' = \{\mathcal{Q}_1', \mathcal{Q}_2\}$ που μπορεί να υπολογιστεί από όλα τα σύγχρονα συστήματα επανεγγραφής. Παρατηρούμε όμως, ότι χρησιμοποιώντας τη μέθοδο που περιγράψαμε παραπάνω δεν μπορούμε να πάρουμε την $\mathcal{R}''$ από την $\mathcal{R}'$. Κάτι τέτοιο είναι δυνατό μόνο με χρήση της $\mathcal{R}$ που είναι κλειστή ως προς το συμπερασμό. $\diamondsuit$

Ο Αλγόριθμος 6 παρουσιάζει την προσέγγιση μας για τον υπολογισμό μιας επανεγγραφής για ένα ερώτημα που επεκτείνεται με ένα νέο άτομο εκμεταλλευόμενος

Αλγόριθμος 6 ExtendRewritingForNewAtoms$(\mathcal{Q}, \mathcal{T}, \mathcal{R}, \rightsquigarrow_{\mathcal{R}}, \alpha)$

Είσοδος: Ένα ΣΕ $\mathcal{Q}$, μια επανεγγραφή $\mathcal{R}$ με διαμέριση $\mathcal{R}_D \uplus \mathcal{R}_Q$, και μια σχέση $\rightsquigarrow_{\mathcal{R}}$ που αποτελούν την έξοδο ενός αλγορίθμου επανεγγραφής για το $\mathcal{Q}$ με βάση ένα TBox $\mathcal{T}$ και ένα άτομο α τ.ω. $\mathsf{var}(\alpha) \subseteq \mathsf{var}(\mathcal{Q})$.

1: Αρχικοποίησε μια επανεγγραφή $\mathcal{R}' := \mathcal{R}_D$ και μια σχέση $\rightsquigarrow_{\mathcal{R}'} := \rightsquigarrow_{\mathcal{R}}$
2: **for all** $\mathcal{Q}_i \in \mathcal{R}_Q$ **do**
3: $\quad \tau := \{x \mapsto y \mid x \in \mathsf{var}(\mathcal{Q}), y \in \mathsf{var}(\mathcal{Q}_i)$ και $x \rightsquigarrow_{\mathcal{R}} y\}$
4: $\quad$ **if** $\mathsf{var}(\alpha\tau) \subseteq \mathsf{var}(\mathcal{Q}_i)$ **then**
5: $\quad\quad \mathcal{Q}'_i := \bigwedge \mathcal{Q}_i \cup \{\alpha\tau\} \rightarrow \mathsf{hd}(\mathcal{Q}_i)$
6: $\quad\quad$ Προσέθεσε το $\mathcal{Q}'_i$ στην $\mathcal{R}'$
7: $\quad\quad$ **if** $\mathsf{needsRewriting}(\mathcal{Q}'_i, \alpha\tau, \mathcal{T})$ **then**
8: $\quad\quad\quad \langle \mathcal{R}_n, \rightsquigarrow_{\mathcal{R}_n} \rangle := \mathsf{rew}(\mathcal{Q}'_i, \mathcal{T})$
9: $\quad\quad\quad \mathcal{R}' := \mathcal{R}' \cup \mathcal{R}_n$ και $\rightsquigarrow_{\mathcal{R}'} := \rightsquigarrow_{\mathcal{R}'} \cup \rightsquigarrow_{\mathcal{R}_n}$
10: $\quad\quad$ **end if**
11: $\quad$ **end if**
12: **end for**
13: **return** $\langle \mathcal{R}', \rightsquigarrow_{\mathcal{R}'} \rangle$

την πληροφορία που έχει παραχθεί προηγούμενα. Ο αλγόριθμος δέχεται ως είσοδο ένα ΣΕ $\mathcal{Q}$, ένα TBox $\mathcal{T}$, μια επανεγγραφή $\mathcal{R} = \mathcal{R}_D \uplus \mathcal{R}_Q$ που είναι κλειστή ως προς το συμπερασμό και μια σχέση $\rightsquigarrow_{\mathcal{R}}$ που αποτελούν την έξοδο ενός αλγορίθμου επανεγγραφής για τα $\mathcal{Q}, \mathcal{T}$, και ένα άτομο α τέτοιο ώστε $\mathsf{var}(\alpha) \subseteq \mathsf{var}(\mathcal{Q})$. Επιπλέον, ο αλγόριθμος μας χρησιμοποιεί εσωτερικά έναν αλγόριθμο επανεγγραφής rew έτσι ώστε να μπορεί να εφαρμόζει τον κανόνα συμπερασμού όπου αυτό κρίνεται απαραίτητο.

Αρχικά, ο αλγόριθμος μας αρχικοποιεί μια νέα επανεγγραφή $\mathcal{R}'$ με το σύνολο $\mathcal{R}_D$ και τη σχέση $\rightsquigarrow_{\mathcal{R}'}$ με την $\rightsquigarrow_{\mathcal{R}}$. Στη συνέχεια, ο αλγόριθμος εισέρχεται στον κεντρικό βρόχο. Προσπελαύνει κάθε ερώτημα $\mathcal{Q}_i \in \mathcal{R}_Q$ και παράγει κατάλληλες αντιστοιχήσεις τ των μεταβλητών του $\mathcal{Q}_i$ μ.β.τ. τις μεταβλητές του $\mathcal{Q}$ (γραμμή 3) με χρήση των οποίων ελέγχει εάν το α έχει κοινές μεταβλητές με το $\mathcal{Q}_i$. Εάν κάτι τέτοιο ισχύει προσθέτει το επεκτεταμένο ερώτημα $\bigwedge \mathsf{bd}(\mathcal{Q}_i) \cup \{\alpha\tau\} \rightarrow \mathsf{hd}(\mathcal{Q}_i)$ στο τελικό αποτέλεσμα. Τέλος χρησιμοποιεί τη συνάρτηση needsRewriting για να ελέγξει εάν το $\mathcal{Q}'_i$ χρειάζεται να επανεγγραφεί περισσότερο και εάν κάτι τέτοιο είναι απαραίτητο προσθέτει την παραγόμενη επανεγγραφή στο αποτέλεσμα (γραμμές 7–10).

Θεώρημα 3.4.3. *Έστω $\mathcal{Q}$ ΣΕ, έστω $\mathcal{T}$ TBox, έστω α άτομο τ.ω.* $\mathsf{var}(\alpha) \in \mathsf{var}(\mathcal{Q})$ *και έστω $\mathcal{R}, \rightsquigarrow_{\mathcal{R}}$ έξοδος ενός αλγόριθμου επανεγγραφής όταν εφαρμόζεται για τα $\mathcal{Q}, \mathcal{T}$ έτσι ώστε η $\mathcal{R}$ να είναι κλειστή ως προ το συμπερασμό. Όταν εφαρμόζεται ο Αλγόριθμος 6 στα $\mathcal{Q}, \mathcal{T}, \mathcal{R}, \rightsquigarrow_{\mathcal{R}}$, και α τερματίζει. Έστω $\mathcal{R}'$ η έξοδος του αλγορίθμου. Τότε το σύνολο $\mathcal{R}'$ είναι μια επανεγγραφή κλειστή ως προς το συμπερασμό για το ερώτημα* $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q})$ *μ.β.τ. $\mathcal{T}$.*

Απόδειξη: Ο τερματισμός του αλγορίθμου προκύπτει από το γεγονός ότι ο Αλγόριθμος 6 προσπελαύνει μόνο μια φορά κάθε ερώτημα που είναι μέλος ενός πεπερασμένου συνόλου ($\mathcal{R}_Q$) καθώς και επειδή ο αλγόριθμος επανεγγραφής rew τερ-

ματίζει.

Έστω, $\mathcal{T}$, $\langle \mathcal{R}, \rightsquigarrow_{\mathcal{R}} \rangle$, $\mathcal{Q}$, και α είσοδος για τον Αλγόριθμο 6, όπου $\langle \mathcal{R}, \rightsquigarrow_{\mathcal{R}} \rangle$ είναι επανεγγραφη που υπολογίζεται από κάποιο αλγόριθμο rew και έστω $\mathsf{var}(\alpha) \subseteq \mathsf{var}(\mathcal{Q})$.

Έστω ότι η έξοδος του Αλγόριθμου 6 είναι ένα ζευγάρι $\langle \mathcal{R}', \rightsquigarrow_{\mathcal{R}'} \rangle$ και έστω επίσης ότι $\langle \mathcal{R}_i, \rightsquigarrow_{\mathcal{R}_i} \rangle$ είναι η επανεγγραφή που υπολογίζεται ύστερα από i βήματα του αλγορίθμου επανεγγραφής rew όταν αυτός εφαρμόζεται για τα $\mathcal{Q}' = \bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q})$ και $\mathcal{T}$.

Η απόδειξη του παραπάνω θεωρήματος προκύπτει από την παρακάτω ιδιότητα, την οποία θα δείξουμε χρησιμοποιώντας επαγωγή πάνω στον αριθμό βημάτων i.

> (AE): Για κάθε $i \geq 0$ και $\mathcal{Q}'_l \in \mathcal{R}_i$ υπάρχει $\mathcal{Q}_l \in \mathcal{R}$ τέτοιο ώστε εάν για την αντικατάσταση τ που ορίζεται από τη γραμμή 3 του Αλγορίθμου 6 ισχύει $\mathsf{var}(\alpha\tau) \subseteq \mathsf{var}(\mathcal{Q}_l)$ και $\mathcal{Q}^e_l = \bigwedge \mathcal{Q}_l \cup \{\alpha\tau\} \rightarrow \mathsf{hd}(\mathcal{Q}_l)$ είναι η επέκταση του $\mathcal{Q}_l$ με το α, ισχύει μια από τις παρακάτω συνθήκες:
>
> 1. το $\mathcal{Q}^e_l$ υπάγει το $\mathcal{Q}'_l$, ή
> 2. η $\mathsf{needsRewriting}(\mathcal{Q}^e_l, \alpha, \mathcal{T})$ επιστρέφει αληθές και κάποιο $\mathcal{Q}''_l \in \mathsf{rew}(\mathcal{Q}^e_l, \mathcal{T})$ υπάγει το $\mathcal{Q}'_l$.

Βασική Περίπτωση ($i = 0$): Αρχικά $\mathcal{R}_0 = \{\mathcal{Q}'\}$ όπου $\mathcal{Q}' = \bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q})$, συνεπώς, η Ιδιότητα (AE) ικανοποιείται.

Βήμα Επαγωγής: Έστω ότι η Ιδιότητα (AE) ισχύει ύστερα από i βήματα για κάθε $\mathcal{Q}'_l \mathcal{R}_i$—δηλαδή, ισχύει είτε το 1. είτε το 2. Στη συνέχεια, έστω ότι στο βήμα $i+1$ ο αλγόριθμος rew εφαρμόζει ένα βήμα συμπερασμού σε κάποιο ερώτημα $\mathcal{Q}'_l$ παράγοντας το $\mathcal{Q}'_{l+1}$. Στη συνέχεια εξετάζουμε κάθε περίπτωση του βήματος επαγωγής ξεχωριστά:

1. Το $\mathcal{Q}^e_l$ υπάγει το $\mathcal{Q}'_l$. Από τις ιδιότητες της υπαγωγής θα ισχύει ότι είτε το $\mathcal{Q}^e_l$ υπάγει το $\mathcal{Q}'_{l+1}$, ή ότι ένας συμπερασμός είναι εφαρμόσιμος στο $\mathcal{Q}^e_l$ και το αποτέλεσμα, έστω $\mathcal{Q}^e_0$ υπάγει το $\mathcal{Q}'_{l+1}$. Εάν ισχύει η πρώτη περίπτωση τότε όντως υπάρχει ερώτημα $\mathcal{Q}_l \in \mathcal{R}$ τέτοιο ώστε το $\mathcal{Q}^e_l$ να υπάγει το $\mathcal{Q}'_{l+1}$. Υποθέτουμε τώρα ότι ισχύει η δεύτερη περίπτωση, ότι δηλαδή, το $\mathcal{Q}^e_0$ υπάγει το $\mathcal{Q}'_{l+1}$. Έστω ότι το ερώτημα $\mathcal{Q}^e_0$ παράγεται από κάποιο κανόνα συμπερασμού στο $\mathcal{Q}^e_l$ που δεν αφορά το άτομο α. Τότε ο κανόνας αυτός μπορεί να είναι εφαρμόσιμος και στο $\mathcal{Q}_l$ και θα έχει ως αποτέλεσμα την παραγωγή του $\mathcal{Q}_0$. Στην περίπτωση αυτή θα ισχύει ότι το $\mathcal{Q}^e_0 = \bigwedge \mathcal{Q}_0 \cup \{\alpha\tau\} \rightarrow \mathsf{hd}(\mathcal{Q}_l)$, για κατάλληλη αντιστοίχιση τ. Στην περίπτωση τώρα που ο κανόνας συμπερασμού εφαρμόζεται στο άτομο α, αυτό συνεπάγεται ότι αυτός ο κανόνας συμπερασμού δεν θα μπορεί να εφαρμοστεί στο $\mathcal{Q}_l$ για την παραγωγή του $\mathcal{Q}_0$ αλλά θα ενεργοποιήσει τη συνάρτηση $\mathsf{needsRewriting}(\mathcal{Q}^e_l, \alpha, \mathcal{T})$ και συνεπώς θα έχουμε $\mathcal{Q}^e_0 \in \mathsf{rew}(Q^e_l, \mathcal{T})$. Συνεπώς, σε κάθε περίπτωση η Ιδιότητα (AE) ισχύει.

2. Η $\mathsf{needsRewriting}(\mathcal{Q}^e_l, \alpha, \mathcal{T})$ επιστρέφει αληθές και υπάρχει ερώτημα $\mathcal{Q}''_l$ τ.ω. το $\mathcal{Q}''_l \in \mathsf{rew}(\mathcal{Q}^e_l, \mathcal{T})$ να υπάγει το $\mathcal{Q}'_l$. Και πάλι, είτε το $\mathcal{Q}''_l$ υπάγει το $\mathcal{Q}'_{l+1}$ ή μπορεί να εφαρμοστεί ένας συμπερασμός στο $\mathcal{Q}''_l$ και το αποτέλεσμα, έστω $\mathcal{Q}''_0$ υπάγει το $\mathcal{Q}''_l$. Σε κάθε περίπτωση το 2. θα ισχύει καθώς θα έχουμε είτε $\mathcal{Q}''_l \in \mathsf{rew}(\mathcal{Q}^e_l, \mathcal{T})$ ή $\mathcal{Q}''_0 \in \mathsf{rew}(\mathcal{Q}^e_l, \mathcal{T})$.

Έστω ότι όταν ένας αλγόριθμος επανεγγραφής rew εφαρμόζεται στα $\mathcal{Q}', \mathcal{T}$, τερματίζει ύστερα από n βήματα παράγοντας την επανεγγραφή $\mathcal{R}_n$. Η Ιδιότητα (AE) συνεπάγεται ότι κατά τον τερματισμό του Αλγορίθμου 6 για κάθε ερώτημα $\mathcal{Q}_l \in \mathcal{R}_n$ θα υπάρχει ένα ερώτημα $\mathcal{Q}'$ στο $\mathcal{R}'$ το οποίο θα υπάγει το $\mathcal{Q}_l$. □

Αξίζει να σημειωθεί ότι όσον αφορά στη χρήση της υπο-ρουτίνας rew και εδώ ισχύουν όσα έχουν συζητηθεί στην ενότητα 3.1.

3.5 Βελτιστοποιήσεις

Εκτός από τον υπολογισμό μιας επανεγγραφής, ιδιαίτερα σημαντική όσον αφορά το πρόβλημα της απάντησης ερωτημάτων είναι η αποτίμηση των ερωτημάτων σε μια (επαγωγική) βάση δεδομένων. Όπως δηλώνεται και από τα Θεωρήματα 3.2.5, 3.3.7 και 3.4.3, οι Αλγόριθμοι 4, 5 και 6 επιστρέφουν μια επανεγγραφή $\mathcal{R}'$ που είναι κλειστή ως προς το συμπερασμό, κάτι το οποίο συνεπάγεται ότι η $\mathcal{R}'$ ενδεχομένως περιέχει περιττά ερωτήματα. Όπως έχει εξηγηθεί προηγούμενα τα επιπλέον αυτά ερωτήματα είναι σημαντικά εάν σκοπεύουμε να καλέσουμε πάλι τους αλγορίθμους αυτούς για να προσθέσουμε περισσότερες διακεκριμένες μεταβλητές ή να προσθέσουμε/αφαιρέσουμε και άλλα άτομα από το ερώτημα. Παρόλα αυτά, όταν θέλουμε να αποτιμήσουμε την $\mathcal{R}'$ σε μια βάση δεδομένων, είναι προτιμότερο να υπολογίσουμε μια ελάχιστη επανεγγραφή $\mathcal{R}''$ που να είναι ισοδύναμη με την $\mathcal{R}'$. Αυτό μπορεί να πραγματοποιηθεί εφαρμόζοντας τον πολύ γνωστό αλγόριθμο απαλοιφής περιττών ερωτημάτων removeRedundant($\mathcal{R}'$), ο οποίος φαίνεται στον Αλγόριθμο 1 (ενότητα 2.3) και ελέγχει εάν ένα ερώτημα $\mathcal{Q}_i \in \mathcal{R}'$ υπάγειται από κάποιο άλλο ερώτημα $\mathcal{Q}_j \in \mathcal{R}'$ (και αντίστροφα) και στη συνέχεια αφαιρεί το κατάλληλο ερώτημα από την τελική επανεγγραφή [116]. Στη διατριβή αυτή δεν ασχολούμαστε με άλλες τεχνικές που στηρίζονται στην δομή και το σχεδιασμό της βάσης δεδομένων [127, 130]. Οι τεχνικές αυτές είναι ανεξάρτητες από τον αλγόριθμο επανεγγραφής και μπορούν να χρησιμοποιηθούν και για τους αλγορίθμους μας.

Συνεπώς, πριν αποτιμήσουμε την επανεγγραφή που επιστρέφεται από τους Αλγορίθμους 4, 5 και 6 μπορούμε να χρησιμοποιήσουμε τον Αλγόριθμο 1 για να απαλείψουμε περιττά ερωτήματα. Όπως έχει δειχθεί όμως [116, 40] η μέθοδος αυτή συνήθως δεν αποδίδει καλά στην πράξη καθώς αποτελείται από δύο εμφωλευμένους βρόχους επανάληψης πάνω σε μια πιθανότατα μεγάλη επανεγγραφή $\mathcal{R}'$. Για να βελτιστοποιήσουμε την απόδοση του Αλγορίθμου 1 οι αλγόριθμοι μας χρησιμοποιούν μια σειρά από προσεγγίσεις με τις οποίες προσπαθούν να μειώσουν το μέγεθος των συνόλων τα οποία θα παράγουν κατά την εφαρμογή τους.

Πρώτα, προσπαθούν να αναγνωρίσουν κατά τη διάρκεια του τρεξίματος τους, ερωτήματα που εάν παραχθούν και προστεθούν στο σύνολο $\mathcal{R}'$ πρόκειται να είναι περιττά. Όσο περισσότερα περιττά ερωτήματα αναγνωριστούν τόσο μικρότερο θα είναι το μέγεθος του συνόλου $\mathcal{R}'$ για το οποίο θα πρέπει να εφαρμοστεί ο Αλγόριθμος 1. Παρόλα αυτά, το $\mathcal{R}'$ μπορεί και πάλι να είναι πολύ μεγάλο. Συνεπώς, κατά δεύτερον, προσπαθούν να αναγνωρίσουν ερωτήματα που πρόκειται να είναι μη-περιττά στο σύνολο $\mathcal{R}'$ που υπολογίζεται. Τέτοια ερωτήματα μπορούν στη συνέχεια να αποκλειστούν

από τον τελικό έλεγχο μειώνοντας το μέγεθος του πρώτου βρόχου της μεθόδου removeRedundant. Πιο συγκεκριμένα, εάν NR είναι το σύνολο των μη-περιττών ερωτημάτων τότε χρειάζεται να ελεγχθούν μόνο τα ΣΕ από το σύνολο $\mathcal{R}' \setminus NR$ σε σχέση με τα ΣΕ από το $\mathcal{R}'$. Τέλος, προσπαθούν να αναγνωρίσουν ερωτήματα που είναι μη-υπάγοντα. Αυτά μπορούν να χρησιμοποιηθούν στη συνέχεια για να μειώσουν το μέγεθος του δεύτερου βρόχου. Πιο συγκρεκριμένα, εάν το NS περιέχει όλα αυτά τα ερωτήματα τότε κάθε ΣΕ στο σύνολο $\mathcal{R} \setminus NS$ πρέπει να ελεγχθεί μόνο με τα ΣΕ $\mathcal{R} \setminus NR$.

Στη συνέχεια παρουσιάζουμε τις προσεγγίσεις αυτές για κάθε έναν από τους Αλγορίθμους 4 και 5.

3.5.1 Βελτιστοποιώντας τον Αλγόριθμο Επέκτασης Ερωτημάτων με Διακεκριμένες Μεταβλητές

Στην ενότητα αυτή παρουσιάζουμε μια σειρά από προτάσεις με χρήση των οποίων μπορούμε να βελτιστοποιήσουμε την εκτέλεση του αλγορίθμου απαλοιφής περιττών ερωτημάτων (Αλγόριθμος 1) για τον Αλγόριθμο 4.

Απαλοίφοντας Περιττά Ερωτήματα

Η παρακάτω πρόταση παρουσιάζει μια ιδιότητα με χρήση της οποίας μπορούμε να αναγνωρίσουμε εάν ένα ερώτημα που κατασκευάζεται κατά την εκτέλεση του Αλγορίθμου 4 θα είναι περιττό στην τελική επανεγγραφή $\mathcal{R}'$.

Πρόταση 3.5.1. *Έστω $\mathcal{T}$ TBox, έστω $\mathcal{Q}$ ΣΕ, έστω $\mathcal{R} = \mathcal{R}_D \uplus \mathcal{R}_Q$ επανεγγραφή για τα $\mathcal{Q}$ και $\mathcal{T}$, και έστω $\mathcal{R}'$ έξοδος του Αλγόριθμου 4 όταν αυτός εκτελείται για την $\mathcal{R}$ και κάποια αυθαίρετη πλειάδα μεταβλητών $\vec{y}$ τ.ω. $\vec{y} \subseteq \mathsf{var}(\mathcal{Q})$. Έστω ΣΕ $\mathcal{Q}_1 \in \mathcal{R}$ που υπάγεται από κάποιο ΣΕ $\mathcal{Q}_2 \in \mathcal{R}$. Τότε, εάν οι επεκτάσεις τους με την $\vec{y}$, $\mathcal{Q}'_1$ και $\mathcal{Q}'_2$ αντίστοιχα, είναι ασφαλείς, το ΣΕ $\mathcal{Q}'_1$ είναι περιττό στην $\mathcal{R}'$.*

Η ισχύς της Πρότασης 3.5.1 είναι ξεκάθαρη. Η Πρόταση αυτή μπορεί να χρησιμοποιηθεί για την αποφυγή της προσθήκης περιττών ερωτημάτων στο αποτέλεσμα $\mathcal{R}'$ του Αλγορίθμου 4. Παρόλα αυτά, για να μπορέσουμε να ελέγξουμε την ισχύ της ιδιότητας αυτής είναι απαραίτητο να γνωρίζουμε τις σχέσεις υπαγωγής που ισχύουν ανάμεσα στα ΣΕ της επανεγγραφής εισόδου $\mathcal{R}$, κάτι το οποίο μπορεί επίσης (όπως και η επανεγγραφή $\mathcal{R}$) να έχει υπολογιστεί προηγούμενα με χρήση του Αλγορίθμου 1. Στη συνέχεια, ο Αλγόριθμος 4 χρησιμοποιεί την Πρόταση 3.5.1 έτσι ώστε να μην προσθέσει την ασφαλή επέκταση $\mathcal{Q}'_i$ ενός ερωτήματος $\mathcal{Q}_i \in \mathcal{R}$ στο σύνολο $\mathcal{R}'$ (γραμμή 5) εάν έχει ήδη προστεθεί στο $\mathcal{R}'$ η ασφαλής επέκταση κάποιου ΣΕ $\mathcal{Q}_j$ το οποίο υπάγει το $\mathcal{Q}_i$.

Παράδειγμα 3.5.2. Έστω ότι για κάποιο ΣΕ και TBox υπάρχει επανεγγραφή $\mathcal{R}$ που περιέχει τα ερωτήματα $\mathcal{Q}_1 = R(x_1, y_1) \wedge S(y_1, z_1) \rightarrow \mathcal{Q}_A(x_1)$ και $\mathcal{Q}_2 = R(x_2, y_2) \rightarrow \mathcal{Q}_A(x_2)$, με $\rightsquigarrow_{\mathcal{R}} = \{\langle x_1, x_2\rangle, \langle y_1, y_2\rangle, \langle z_1, y_2\rangle\}$. Είναι προφανές ότι το $\mathcal{Q}_2$ υπάγει το $\mathcal{Q}_1$. Έστω τώρα ότι καλείται ο Αλγόριθμος 4 για την μεταβλητή y_1. Είναι εύκολο να παρατηρήσει κανείς πως η επέκταση $\mathcal{Q}'_2$ του $\mathcal{Q}_2$ με την μεταβλητή y_1 υπάγει

την επέκταση $\mathcal{Q}_1'$ του $\mathcal{Q}_1$ με την ίδια μεταβλητή. Έτσι λοιπόν όταν ο Αλγόριθμος 4 προσπελάσει το $\mathcal{Q}_1$ για να το επεκτείνει, στην περίπτωση που η επέκταση $\mathcal{Q}_2'$ του $\mathcal{Q}_2$ έχει ήδη προστεθεί στην τελική επανεγγραφή με χρήση της Πρότασης 3.5.1 θα απορρίψει το $\mathcal{Q}_1$ και δε θα το επεξεργαστεί περαιτέρω. ♢

Εντοπισμός μη-Περιττών Ερωτημάτων

Η παρακάτω πρόταση παρουσιάζει μια ιδιότητα με χρήση της οποίας μπορούμε να αναγνωρίσουμε εάν ένα ερώτημα που κατασκευάζεται κατά την εκτέλεση του Αλγορίθμου 4 θα είναι μη-περιττό στην τελική επανεγγραφή $\mathcal{R}'$.

Πρόταση 3.5.3. *Έστω $\mathcal{T}$ TBox, $\mathcal{Q}$ ΣΕ, έστω $\mathcal{R} = \mathcal{R}_D \uplus \mathcal{R}_Q$ επανεγγραφή για τα $\mathcal{Q}$ και $\mathcal{T}$, και έστω $\mathcal{R}'$ έξοδος του Αλγόριθμου 4 όταν αυτός εκτελείται για την $\mathcal{R}$ και κάποια αυθαίρετη πλειάδα μεταβλητών $\vec{y}$ τ.ω. $\vec{y} \subseteq \mathsf{var}(\mathcal{Q})$. Έστω ΣΕ $\mathcal{Q}_1 \in \mathcal{R}$ που είναι μη-περιττό στην $\mathcal{R}$ και έστω $\mathcal{Q}_1'$ η επέκταση του με την $\vec{y}$. Εάν το $\mathcal{Q}_1'$ είναι ασφαλές τότε θα είναι και μη-περιττό στην $\mathcal{R}'$.*

Η ισχύς και αυτής της πρότασης είναι ξεκάθαρη. Για να μπορεί να χρησιμοποιηθεί η Πρόταση 3.5.3 ο Αλγόριθμος 4 τροποποιείται ως εξής:

- Στην αρχή αρχικοποιεί ένα άδειο σύνολο NR που περιέχει μη-περιττά ερωτήματα.
- Στη γραμμή 5, εάν το $\mathcal{Q}_i$ είναι μη-περιττό στην $\mathcal{R}$ προσθέτει το $\mathcal{Q}_i'$ στο NR.
- Τέλος επιστρέφει την $\mathcal{R}$ και το σύνολο NR.

Στη συνέχεια, το σύνολο NR που έχει επιστραφεί χρησιμοποιείται από τον Αλγόριθμο 1 ως εξής: Όλα τα ερωτήματα που ανήκουν στο σύνολο NR αποκλείονται από τον έλεγχο απαλοιφής όπως φαίνεται και στον Αλγόριθμο 7.

Παράδειγμα 3.5.4. Έστω το ΣΕ, TBox και η επανεγγραφή $\mathcal{R}$ από το Παράδειγμα 3.5.2. Εάν η $\mathcal{R}$ δεν περιέχει άλλα ερωτήματα τότε εφόσον η επέκταση $\mathcal{Q}_1'$ του $\mathcal{Q}_1$ είναι ασφαλής οι συνθήκες της Πρότασης 3.5.3 ικανοποιούνται και συνεπώς το $\mathcal{Q}_1'$ είναι μη-περιττό στην τελική επανεγγραφή $\mathcal{R}'$ που προκύπτει από την εφαρμογή του Αλγορίθμου 4 για την μεταβλητή y_1. ♢

Εντοπισμός μη-Υπαγόντων Ερωτημάτων

Τέλος, η παρακάτω πρόταση παρουσιάζει μια ιδιότητα την οποία μπορούμε να χρησιμοποιήσουμε για να αναγνωρίσουμε εάν ένα ερώτημα που κατασκευάζεται κατά την εκτέλεση του Αλγορίθμου 4 θα είναι μη-υπάγον στην τελική επανεγγραφή $\mathcal{R}'$.

Πρόταση 3.5.5. *Έστω $\mathcal{T}$ TBox, $\mathcal{Q}$ ΣΕ, έστω $\mathcal{R} = \mathcal{R}_D \uplus \mathcal{R}_Q$ επανεγγραφή για τα $\mathcal{Q}$ και $\mathcal{T}$, και έστω $\mathcal{R}'$ έξοδος του Αλγόριθμου 4 όταν αυτός εκτελείται για την $\mathcal{R}$ και κάποια αυθαίρετη πλειάδα μεταβλητών $\vec{y}$ τ.ω. $\vec{y} \subseteq \mathsf{var}(\mathcal{Q})$. Έστω ΣΕ $\mathcal{Q}_1 \in \mathcal{R}$ που είναι μη-υπάγον στην $\mathcal{R}$ και έστω $\mathcal{Q}_1'$ η επέκταση του με την $\vec{y}$. Εάν το $\mathcal{Q}_1'$ είναι ασφαλές τότε θα είναι και μη-υπάγον στην $\mathcal{R}'$.*

Αλγόριθμος 7 removeRedundantOpt($\mathcal{R}, NS, NR$)

Είσοδος: Μια επανεγγραφή $\mathcal{R} = \mathcal{R}_D \uplus \mathcal{R}_Q$.

1: Αρχικοποίησε μια επανεγγραφή $\mathcal{R}' = \mathcal{R}$
2: Αρχικοποίησε μια επανεγγραφή ΣΣΕ $\mathcal{R}_{NS} = \mathcal{R} \setminus NS$
3: Αρχικοποίησε μια επανεγγραφή ΣΣΕ $\mathcal{R}_{NR} = \mathcal{R} \setminus NR$
4: **for all** $\mathcal{Q} \in \mathcal{R}_{NS}$ **do**
5: **for all** $\mathcal{Q}' \in \mathcal{R}_{NR}$ με $\mathcal{Q} \neq \mathcal{Q}'$ **do**
6: **if** το $\mathcal{Q}$ υπάγει το $\mathcal{Q}'$ και $\mathcal{Q} \in \mathcal{R}'$ **then**
7: Αφαίρεσε το $\mathcal{Q}'$ από το $\mathcal{R}'$
8: **end if**
9: **end for**
10: **end for**
11: **return** $\mathcal{R}'$

Η ισχύς και αυτής της πρότασης είναι ξεκάθαρη. Όμοια με πριν, ο Αλγόριθμος 4 χρησιμοποιεί επιπλέον σύνολα (το NS) για να αποθηκεύσει τέτοια ερωτήματα τα οποία στη συνέχεια εξαιρούνται από τον βρόχο της συνάρτησης removeRedundant όπως φαίνεται και στον Αλγόριθμο 7.

Παράδειγμα 3.5.6. Έστω το ΣΕ, TBox από το Παράδειγμα 3.5.2 και έστω ότι η επανεγγραφή $\mathcal{R}$ περιέχει επίσης το ερώτημα $\mathcal{Q}_3 = K(x_3, y_3) \rightarrow \mathcal{Q}_A(x_3)$. Τότε, εφόσον το $\mathcal{Q}_3$ δεν υπάγει κανένα ερώτημα στην $\mathcal{R}$ και επειδή η επέκταση του $\mathcal{Q}'_3 = K(x_3, y_3) \rightarrow \mathcal{Q}_A(x_3, y_3)$ είναι ασφαλής το ερώτημα $\mathcal{Q}'_3$ θα είναι μη-υπάγον στην τελική επανεγγραφή $\mathcal{R}'$. ♢

3.5.2 Βελτιστοποιώντας τον Αλγόριθμο Μείωσης Ερωτημάτων με Άτομα

Στην ενότητα αυτή παρουσιάζουμε μια σειρά από προτάσεις που έχουν ως στόχο τη βελτιστοποίηση της εκτέλεσης του Αλγορίθμου 1 όταν αυτός εκτελείται για τον Αλγόριθμο 5.

Απαλοιφή Περιττών Ερωτημάτων

Η παρακάτω πρόταση παρουσιάζει δύο ιδιότητες με χρήση των οποίων μπορούμε να αναγνωρίσουμε εάν ένα ερώτημα που κατασκευάζεται κατά την εκτέλεση του Αλγόριθμου 5 θα είναι περιττό στην τελική επανεγγραφή $\mathcal{R}'$.

Πρόταση 3.5.7. *Έστω $\mathcal{Q}$ ΣΕ, έστω $\mathcal{T}$ TBox, έστω $\langle \mathcal{R}, \rightsquigarrow_{\mathcal{R}} \rangle$ έξοδος ενός αλγορίθμου επανεγγραφής όταν εκτελείται για τα $\mathcal{Q}, \mathcal{T}$, και έστω α άτομο του $\mathcal{Q}$. Έστω επίσης $\langle \mathcal{R}', \rightsquigarrow_{\mathcal{R}'} \rangle$ έξοδος του Αλγορίθμου 5 όταν εκτελείται για τα $\mathcal{T}, \mathcal{R}, \rightsquigarrow_{\mathcal{R}}$, και α, έστω $\mathcal{Q}_i \in \mathcal{R}$ ΣΕ, και έστω $\tau = \{x \mapsto y \mid x \in \mathsf{var}(\mathcal{Q}), y \in \mathsf{var}(\mathcal{Q}_i)$ και $x \rightsquigarrow_{\mathcal{R}} y\}$, και έστω ότι ο Αλγόριθμος 5 δεν εκτελεί ποτέ την γραμμή 15. Εάν υπάρχει $\mathcal{Q}_j \in \mathcal{R}$ τ.ω. να υπάγει το $\mathcal{Q}_i$, τότε ισχύουν οι παρακάτω ιδιότητες, όπου $\tau' = \{x \mapsto y \mid x \in \mathsf{var}(\mathcal{Q}), y \in \mathsf{var}(\mathcal{Q}_j)$ και $x \rightsquigarrow_{\mathcal{R}} y\}$:*

(ARR1) *Εάν $\mathcal{Q}_j \in \mathcal{R}'$ ή $\bigwedge \mathsf{bd}(\mathcal{Q}_j) \setminus \{\alpha\tau'\} \rightarrow \mathsf{hd}(\mathcal{Q}) \in \mathcal{R}'$, τότε το $\mathcal{Q}_i$ είναι περιττό στην $\mathcal{R}'$,*

(ARR2) *Εάν $\mathcal{Q}_j \in \mathcal{R}'$ ($\bigwedge \mathsf{bd}(\mathcal{Q}_j) \setminus \{\alpha\tau'\} \rightarrow \mathsf{hd}(\mathcal{Q}) \in \mathcal{R}'$) και για κάθε $\alpha' \in \mathsf{bd}(\mathcal{Q}_j)$ ($\alpha' \in \mathsf{bd}(\mathcal{Q}_j) \setminus \{\alpha\tau'\}$) το άτομο α' έχει διαφορετικό κατηγόρημα από το α τότε το ερώτημα $\bigwedge \mathsf{bd}(\mathcal{Q}_i) \setminus \{\alpha\tau\} \rightarrow \mathsf{hd}(\mathcal{Q})$ είναι περιττό.*

Απόδειξη: (Ιδιότητα (ARR1)) Εφόσον το $\mathcal{Q}_i$ υπάγειται από το $\mathcal{Q}_j$ υπάρχει αντικατάσταση θ τέτοια ώστε για κάθε $At \in \mathcal{Q}_j\theta$ να έχουμε $At \in \mathcal{Q}_i$. Τώρα, η Ιδιότητα (ARR1) προκύπτει ευθέως εφόσον για την ίδια αντικατάσταση θ για την οποία για κάθε $At \in \mathcal{Q}_j\theta$ (ή $At \in [\bigwedge \mathsf{bd}(\mathcal{Q}_j) \setminus \{\alpha\tau'\} \rightarrow \mathsf{hd}(\mathcal{Q})]\theta$) θα έχουμε $At \in \mathcal{Q}_i$.

(Ιδιότητα (ARR2)) Έστω ότι $\mathcal{Q}_j \in \mathcal{R}'$ (η περίπτωση με το $\bigwedge \mathsf{bd}(\mathcal{Q}_j) \setminus \{\alpha\tau'\} \in \mathcal{R}'$ είναι παρόμοια). Από την υπόθεση έχουμε ότι για κάθε $At \in \mathcal{Q}_j\theta$ έχουμε επίσης $At \in \mathcal{Q}_i$. Έστω ένα αυθαίρετο άτομο At. Μπορούμε να έχουμε $At \notin \mathsf{bd}(\mathcal{Q}_i) \setminus \{\alpha\tau\}$ εάν $At = \alpha\tau$, δηλαδή, το άτομο που αφαιρείται είναι κάποιο άτομο από το $\mathcal{Q}_j\theta$. Από την υπόθεση όμως το At έχει διαφορετικό κατηγόρημα από το α, συνεπώς θα έχουμε ότι $At \neq \alpha\tau$ για κάθε τ. □

Παρόμοια με πριν, η Πρόταση 3.5.7 μπορεί να χρησιμοποιηθεί για την αποφυγή προσθήκης περιττών ερωτημάτων στο αποτέλεσμα $\mathcal{R}'$ του Αλγορίθμου 5. Για να γίνει κάτι τέτοιο ο Αλγόριθμος 5 τροποποιείται ως εξής:

- Στη γραμμή 10 δεν προσθέτει κανένα ερώτημα $\mathcal{Q}_i$ στο $\mathcal{R}'$ εάν το $\mathcal{Q}_i$ υπάγεται από κάποιο ερώτημα $\mathcal{Q}_j \in \mathcal{R}$ και είτε $\mathcal{Q}_j \in \mathcal{R}'$ ή $\bigwedge \mathsf{bd}(\mathcal{Q}_j) \setminus \{\alpha\tau'\} \rightarrow \mathsf{hd}(\mathcal{Q}) \in \mathcal{R}'$.

- Έστω $\mathcal{Q}_i$ ΣΕ που επιλέγεται στη γραμμή 7. Εάν το $\mathcal{Q}_i$ υπάγεται από κάποιο $\mathcal{Q}_j \in \mathcal{R}$ και είτε αυτό είτε η μείωση του έχει προστεθεί στην τελική επανεγγραφή $\mathcal{R}'$ και για κάθε $\alpha' \in \mathsf{bd}(\mathcal{Q}_j)$ ή $\alpha' \in \mathsf{bd}(\mathcal{Q}_j) \setminus \{\alpha\tau'\}$ το άτομο α' έχει διαφορετικό κατηγόρημα από το α τότε «προσπερνάει» τις γραμμές 11–18 και συνεχίζει για το επόμενο ερώτημα.

Παράδειγμα 3.5.8. Έστω ΣΕ και TBox από τα οποία παράγεται επανεγγραφή $\mathcal{R}$ που περιέχει τα ΣΕ $\mathcal{Q}_1 = A(x_1) \rightarrow \mathcal{Q}_A(x_1)$ και $\mathcal{Q}_2 = A(x_2) \wedge R(x_2, y_2) \wedge S(x_2, y_2) \rightarrow \mathcal{Q}_A(x_2)$. Είναι προφανές ότι το $\mathcal{Q}_1$ υπάγει το $\mathcal{Q}_2$ στην $\mathcal{R}$. Έστω τώρα ότι καλείται ο Αλγόριθμος 5 για το άτομο $\alpha = R(x, y)$ και έστω ότι το $\mathcal{Q}_1$ είναι ανεξάρτητο από το α, συνεπώς $\mathcal{Q}_1 \in \mathcal{R}$. Παρατηρούμε ότι τόσο η Ιδιότητα (ARR1) όσο και η Ιδιότητα (ARR2) ικανοποιούνται συνεπώς εάν το $\mathcal{Q}_2$ ή το $\mathcal{Q}_2' = A(x_2) \wedge S(x_2, y_2) \rightarrow \mathcal{Q}_A(x_2)$ προστεθεί στο τελικό αποτέλεσμα $\mathcal{R}'$, τότε αυτό θα είναι περιττό.

Εντοπισμός μη-Περιττών Ερωτημάτων

Η παρακάτω πρόταση παρουσιάζει μεθόδους για την αναγνώριση μη-περιττών ερωτημάτων κατά τη διάρκεια εκτέλεσης του Αλγόριθμου 5 υποθέτοντας ότι γνωρίζουμε τις σχέσεις υπαγωγής από την επανεγγραφή της εισόδου.

Πρόταση 3.5.9. *Έστω $\mathcal{Q}$ ΣΕ, έστω $\mathcal{T}$ TBox, έστω $\langle \mathcal{R}, \rightsquigarrow_{\mathcal{R}} \rangle$ έξοδος ενός αλγορίθμου επανεγγραφής όταν εκτελείται για τα $\mathcal{Q}$ και $\mathcal{T}$, και έστω α άτομο του $\mathcal{Q}$. Έστω επίσης*

$\langle \mathcal{R}', \rightsquigarrow_{\mathcal{R}'} \rangle$ έξοδος του Αλγορίθμου 5 όταν εκτελείται για τα $\mathcal{T}, \mathcal{R}, \rightsquigarrow_{\mathcal{R}}$, και α, έστω $\mathcal{Q}_i \in \mathcal{R}$ ΣΕ, και έστω $\tau = \{x \mapsto y \mid x \in \mathsf{var}(\mathcal{Q}), y \in \mathsf{var}(\mathcal{Q}_i)$ και $x \rightsquigarrow_{\mathcal{R}} y\}$, και έστω ότι ο Αλγόριθμος 5 δεν εκτελεί ποτέ την γραμμή 15. Εάν το $\mathcal{Q}_i$ είναι μη-περιττό στην $\mathcal{R}$ και για κάθε $\mathcal{Q}_j \in \mathcal{R}$ η αντιστοίχιση τ_i που ορίζεται ως $\{x \mapsto y \mid x \in \mathsf{var}(\mathcal{Q}), y \in \mathsf{var}(\mathcal{Q}_j)$ και $x \rightsquigarrow_{\mathcal{R}} y\}$ είναι ένα-προς-ένα, τότε ισχύουν οι παρακάτω ιδιότητες:

(ARN1) *Εάν $\alpha\tau \in \mathsf{bd}(\mathcal{Q}_i)$, τότε το $\mathcal{Q}_i$ είναι μη περιττό στην $\mathcal{R}'$,*

(ARN2) *το $\bigwedge \mathsf{bd}(\mathcal{Q}_i) \setminus \{\alpha\tau\} \rightarrow \mathsf{hd}(\mathcal{Q}_i)$ είναι μη-περιττό στην $\mathcal{R}'$.*

Απόδειξη: (Ιδιότητα (ARN1)) Έστω $\mathcal{Q}_j$ αυθαίρετο ΣΕ στην $\mathcal{R}$ διάφορο του $\mathcal{Q}_i$. Κατά τον τερματισμό του Αλγορίθμου 5 είτε το $\mathcal{Q}_j$ ή το $\bigwedge \mathsf{bd}(\mathcal{Q}_j) \setminus \{\alpha\tau_j\} \rightarrow \mathsf{hd}(\mathcal{Q})$, για μια αντικατάσταση τ_j που αντιστοιχεί τις μεταβλητές του $\mathcal{Q}$ σε αυτές του $\mathcal{Q}_j$, προστίθεται στην $\mathcal{R}'$. Εφόσον το $\mathcal{Q}_i$ είναι μη-περιττό στην $\mathcal{R}$, το $\mathcal{Q}_j$ δεν μπορεί να υπάγει το $\mathcal{Q}_i$. Άρα, πρέπει να δειξουμε ότι και το $\bigwedge \mathsf{bd}(\mathcal{Q}_j) \setminus \{\alpha\tau_j\} \rightarrow \mathsf{hd}(\mathcal{Q})$ δεν υπάγει επίσης το $\mathcal{Q}_i$.

Εφόσον το $\mathcal{Q}_i$ είναι μη-περιττό αυτό σημαίνει ότι για κάθε $\mathcal{Q}_j$ στην $\mathcal{R}$ και κάθε αντικατάσταση θ υπάρχει $At \in \mathcal{Q}_j$ τ.ω. $At\theta \notin \mathcal{Q}_i$. Έστω θ' τέτοια ώστε $\tau = \tau_j\theta'$. Αυτό είναι πιθανό εφόσον η τ_j είναι ένα-προς-ένα, άρα υπάρχει μια αντίστροφη αντικατάσταση τ_j^- και τότε μπορούμε να ορίσουμε $\theta' = \tau_j^-\tau$. Από τη συνθήκη της μη-περιττότητας του $\mathcal{Q}_i$ έχουμε επίσης ότι για την θ' αυτή υπάρχει $At \in \mathcal{Q}_j$ τ.ω. $At\theta' \notin \mathcal{Q}_i$. Έστω τώρα αυτό το συγκεκριμένο At. Αρκεί να δείξουμε ότι $At \in \mathcal{Q}_j \setminus \alpha\tau_j$. Έστω ότι ισχύει το αντίστροφο, δηλαδή ότι $At \notin \mathcal{Q}_j \setminus \alpha\tau_j$. Εφόσον $At \in \mathcal{Q}_j$ αυτό συνεπάγεται ότι $At = \alpha\tau_j$. Επειδή $At\theta' \notin \mathcal{Q}_i$ έχουμε επίσης ότι $\alpha\tau_j\theta' \notin \mathcal{Q}_i$, όμως, λόγω κατασκευής της θ' έχουμε $\alpha\tau \notin \mathsf{bd}(\mathcal{Q}_i)$ που μας οδηγεί σε άτοπο.

(Ιδιότητα (ARN2)) Η ιδιότητα αυτή προκύπτει από την Ιδιότητα (ARN1) επειδή εφόσον το $\bigwedge \mathsf{bd}(\mathcal{Q}_j) \setminus \{\alpha\tau_j\} \rightarrow \mathsf{hd}(\mathcal{Q})$ δεν υπάγει το $\mathcal{Q}_i$ τότε είναι προφανές ότι δεν θα υπάγει επίσης και το $\bigwedge \mathsf{bd}(\mathcal{Q}_i) \setminus \{\alpha\tau\} \rightarrow \mathsf{hd}(\mathcal{Q})$. □

Για να μπορούν να χρησιμοποιηθούν οι Ιδιότητες (ARN1) και (ARN2) ο Αλγόριθμος 5 τροποποιείται ως εξής:

- Στην αρχή αρχικοποιεί ένα άδειο σύνολο NR που περιέχει μη-περιττά ερωτήματα.
- Στην γραμμή 10 εάν το $\mathcal{Q}_i$ περιέχει το άτομο $\alpha\tau$ τότε το προσθέτει στο σύνολο NR.
- Στη γραμμή 12 που κατασκευάζει τη μείωση $\bigwedge \mathsf{bd}(\mathcal{Q}_i) \setminus \{\alpha\tau\} \rightarrow \mathsf{hd}(\mathcal{Q}_i)$ του μη-περιττού ερωτήματος $\mathcal{Q}_i$, την προσθέτει στο σύνολο NR.

Στη συνέχεια, όπως προηγούμενα, το σύνολο NR που έχει επιστραφεί χρησιμοποιείται από τον Αλγόριθμο 1 για να αποκλείσει από τον έλεγχο απαλοιφής όλα τα ερωτήματα που ανήκουν στο NR, όπως φαίνεται και από τον Αλγόριθμο 7.

Παράδειγμα 3.5.10. Έστω ΣΕ και TBox από τα οποία παράγεται επανεγγραφή $\mathcal{R}$ που περιέχει τα ερωτήματα $\mathcal{Q}_1 = A(x_1) \wedge R(x_1, y_1) \rightarrow \mathcal{Q}_A(x_1)$ και $\mathcal{Q}_2 = A(x_2) \wedge R(x_2, y_2) \wedge S(x_2, y_2) \rightarrow \mathcal{Q}_A(x_2)$ με $\rightsquigarrow_{\mathcal{R}} = \{\langle x_1, x_2\rangle, \langle y_1, y_2\rangle\}$. Τότε παρατηρούμε ότι

το $\mathcal{Q}_1$ είναι μη-περιττό στην $\mathcal{R}$. Έστω τώρα ότι εφαρμόζεται ο Αλγόριθμος 5 για το άτομο $\alpha = R(x_1, y_1)$ και η επανεγγραφή που παράγεται είναι η $\mathcal{R}'$. Εάν η $\mathcal{R}$ δεν περιέχει άλλα ερωτήματα τότε το ερώτημα $\mathcal{Q}_1$ ($\mathcal{Q}'_1 = A(x_1) \rightarrow \mathcal{Q}_A(x_1)$) θα είναι μη-περιττό στην τελική επανεγγραφή καθώς η Ιδιότητα (ARN1) (Ιδιότητα (ARN2)) ικανοποιείται. ♢

Εντοπισμός μη-Υπαγόντων Ερωτημάτων

Τέλος, η παρακάτω πρόταση παρουσιάζει μια ιδιότητα με χρήση της οποίας μπορούμε να αναγνωρίσουμε εάν ένα ερώτημα που κατασκευάζεται κατά την εκτέλεση του Αλγορίθμου 5 θα είναι μη-υπάγον στην τελική επανεγγραφή $\mathcal{R}'$.

Πρόταση 3.5.11. *Έστω $\mathcal{Q}$ ΣΕ, έστω $\mathcal{T}$ TBox, έστω $\langle \mathcal{R}, \rightsquigarrow_{\mathcal{R}} \rangle$ έξοδος ενός αλγορίθμου επανεγγραφής όταν εκτελείται για τα $\mathcal{Q}$ και $\mathcal{T}$, και έστω α άτομο του $\mathcal{Q}$. Έστω επίσης $\langle \mathcal{R}', \rightsquigarrow_{\mathcal{R}'} \rangle$ έξοδος του Αλγορίθμου 5 όταν εκτελείται για τα $\mathcal{T}, \mathcal{R}, \rightsquigarrow_{\mathcal{R}}$, και α, έστω $\mathcal{Q}_i \in \mathcal{R}$ ΣΕ, και έστω ότι ο Αλγόριθμος 5 δεν εκτελεί ποτέ την γραμμή 15. Εάν το $\mathcal{Q}_i$ είναι μη-υπάγον στην $\mathcal{R}$ τότε θα είναι μη-υπάγον και στην $\mathcal{R}'$.*

Απόδειξη: Εφόσον το $\mathcal{Q}_i$ δεν υπάγει κανένα ΣΕ στην $\mathcal{R}$, έχουμε ότι $\mathcal{Q}_i\theta \not\subseteq \mathcal{Q}_j$ για όλα τα $\mathcal{Q}_j \in \mathcal{R}$ που είναι διάφορα από το $\mathcal{Q}_i$ και για όλες τις αντικαταστάσεις θ. Συνεπώς, για κάθε θ υπάρχει τουλάχιστον ένα άτομο At στο σώμα του $\mathcal{Q}_i$ για το οποίο $At \in \mathcal{Q}_i\theta$ και $At \notin \mathcal{Q}_j$. Κατά τον τερματισμό του Αλγορίθμου 5 η τελική επανεγγραφή $\mathcal{R}'$ μπορεί να περιέχει είτε το $\mathcal{Q}_j$ ή το $\bigwedge \mathsf{bd}(\mathcal{Q}_j) \setminus \{\alpha\tau'\} \rightarrow \mathsf{hd}(\mathcal{Q}_j)$, για μια αντικατάσταση τ' που αντιστοιχεί τις μεταβλητές του $\mathcal{Q}$ στις μεταβλητές του $\mathcal{Q}_j$. Είναι προφανές ότι και στις δύο περιπτώσεις και για κάθε θ θα υπάρχει ένα άτομο $At \in \mathcal{Q}_i\theta$ τέτοιο ώστε $At \notin \mathcal{Q}_j$ ή $At \notin \bigwedge \mathsf{bd}(\mathcal{Q}_j) \setminus \{\alpha\tau'\} \rightarrow \mathsf{hd}(\mathcal{Q}_j)$. □

Ο Αλγόριθμος 5 χρησιμοποιεί επιπλεόν το σύνολο NS για να αποθηκεύσει τέτοια ερωτήματα τα οποία στη συνέχεια χρησιμοποιούνται από τον Αλγόριθμο 7 για να βελτιώσουν την απόδοση του.

Παράδειγμα 3.5.12. Έστω η επανεγγραφή $\mathcal{R}$ από το Παράδειγμα 3.5.10. Εάν το ερώτημα $\mathcal{Q}_2$ προστεθεί από τον Αλγόριθμο 5 στην $\mathcal{R}'$, εάν δηλαδή είναι ανεξάρτητο από το α, τότε εφόσον είναι μη-υπάγον στην $\mathcal{R}$ θα είναι μη-υπάγον και στην $\mathcal{R}'$. Πράγματι, όποιο από τα ερωτήματα $\mathcal{Q}_1$ ή $\mathcal{Q}'_1$ και να προστεθεί στην $\mathcal{R}'$ το $\mathcal{Q}_2$ δεν πρόκειται να το υπάγει. ♢

Κεφάλαιο 4

Βελτιστοποίηση Επανεγγραφής Ερωτημάτων μέσω της Επέκτασης με Άτομα

Στο προηγούμενο κεφάλαιο μελετήσαμε το πρόβλημα του υπολογισμού μιας επανεγγραφής για ερωτήματα που έχουν τροποποιηθεί με προσθήκη/αφαίρεση διακεκριμένων μεταβλητών ή με προσθήκη/αφαίρεση ατόμων. Οι αλγόριθμοι που παρουσιάσαμε δέχονται ως είσοδο ένα ΣΕ $\mathcal{Q}$, ένα TBox $\mathcal{T}$, μια επανεγγραφή $\mathcal{R}$ για τα $\mathcal{Q}, \mathcal{T}$ και μια μεταβλητή ή ένα άτομο και παράγουν μια επανεγγραφή για το νέο τροποποιημένο ερώτημα, εκμεταλλευόμενοι όσο το δυνατόν περισσότερη από την ήδη υπολογισμένη πληροφορία. Τα TBox για τα οποία μπορούν να εφαρμοστούν οι αλγόριθμοι αυτοί περιέχουν αξιώματα εκφραστικών γλωσσών, την εκφραστικότητα των οποίων καταγράφουν οι υπαρξιακοί κανόνες. Παρόλα αυτά, οι αλγόριθμοι αυτοί (εκτός από τον αλγόριθμο επέκτασης διακεκριμένων μεταβλητών) παρουσιάζουν ένα σημαντικό μειονέκτημα. Η λειτουργία τους στηρίζεται στην χρήση ενός αλγορίθμου επανεγγραφής rew για την παραγωγή αποτελεσμάτων που είναι πλήρη. Κάτι τέτοιο όμως, όπως θα φανεί και από την αξιολόγηση που πραγματοποιήσαμε (κεφάλαιο 6), περιορίζει την αποδοτικότητα τους. Για να αντιμετωπιστεί αυτό πρέπει η κλήση του αλγορίθμου rew να ελαχιστοποιηθεί ει δυνατόν να μην καλείται καθόλου. Η μελέτη κατασκευής αλγορίθμων παραγωγής επανεγγραφών για τροποποιημένα ερωτήματα οι οποίοι δεν χρησιμοποιούν κάποιον αλγόριθμο επανεγγραφής rew είναι ιδιαίτερα δύσκολη για λογικές που φτάνουν στην εκφραστικότητα των υπαρξιακών κανόνων. Για το λόγο αυτό, στα πλαίσια αυτής της διατριβής, προτιμήσαμε να μελετήσουμε το πρόβλημα αυτό για γλώσσες χαμηλότερης εκφραστικότητας.

Έτσι, στο κεφάλαιο αυτό περιορίζουμε την εκφραστικότητα των γλωσσών που μελετάμε σε αυτήν της DL-Lite$_R$ και μελετάμε το πρόβλημα υπολογισμού μιας επανεγγραφής για ερωτήματα που έχουν επεκταθεί με νέα άτομα. Από όλα τα προβλήματα που μελετήσαμε επιλέγουμε να μελετήσουμε το πρόβλημα της επέκτασης ερωτημάτων με άτομα, επειδή παρουσιάζει ιδιαίτερο ενδιαφέρον μιας και συνεπάγεται τη δυνατότητα κατασκευής ενός επαυξητικού αλγορίθμου επανεγγραφής για σταθερά ερωτήματα. Πιο συγκεκριμένα, για μια DL-Lite$_R$ οντολογία, ένα συζευκτικό ερώτημα, μια ήδη υπολογισμένη επανεγγραφή, που αποτελείται μόνο από την επανεγγραφή ΣΣΕ,

για το ερώτημα αυτό και ένα άτομο που θα προστεθεί στο ερώτημα παρουσιάζουμε πώς μπορούμε να υπολογίσουμε ένα σύνολο συζευκτικών ερωτημάτων για το νέο ερώτημα επεκτείνοντας την επανεγγραφή ΣΣΕ που έχουμε σαν είσοδο, αποφεύγοντας να το υπολογίσουμε εξαρχής. Ο αλγόριθμος που παρουσιάζουμε είναι αρκετά σύνθετος και για το λόγο αυτό στην ενότητα 4.1 χρησιμοποιούμε μια σειρά από παραδείγματα για να παρουσιάσουμε μια επισκόπησή του, δίνοντας έμφαση σε διάφορα τεχνικά σημεία. Τα παραδείγματα αυτά, όπως είναι φυσιολογικό παρουσιάζουν επικαλύψεις με τα παραδείγματα που παρουσιάσαμε στην ενότητα 3.4, αλλά για λόγους πληρότητας και ευκολίας κατανόησης του αλγορίθμου μας τα παρουσιάζουμε πάλι, με μικρές τροποποιήσεις. Στη συνέχεια, στην ενότητα 4.2 περιγράφουμε αναλυτικά τον αλγόριθμο επέκτασης μιας επανεγγραφής ΣΣΕ και τέλος αποδεικνύουμε την ορθότητα του (ενότητα 4.2.1).

Αξίζει να σημειωθεί ότι εφόσον οι αλγόριθμοι που παρουσιάζουμε στη συνέχεια αναφέρονται σε οντολογίες εκφραστικότητας DL-Lite για την παραγωγή των διαφόρων επανεγγραφών θα χρησιμοποιήσουμε την ορολογία που παρουσιάστηκε στην ενότητα 2.5.1 και για κάθε επανεγγραφή $\mathcal{R} = \mathcal{R}_D \uplus \mathcal{R}_Q$ θα έχουμε $\mathcal{R}_D = \emptyset$.

4.1 Επισκόπηση Αλγορίθμου

Έστω το παρακάτω DL-Lite$_R$ TBox και το ΣΕ που ανακτά όλα τα άτομα που διδάσκουν κάποιον:

$$\begin{aligned} \mathcal{T} &= \{\textsf{Καθηγητής}(x) \rightarrow \exists y.\textsf{διδάσκει}(x,y),\ \exists y.\textsf{διδάσκει}(y,x) \rightarrow \textsf{Φοιτητής}(x)\} \\ \mathcal{Q} &= \textsf{διδάσκει}(x,y) \rightarrow \mathcal{Q}_A(x) \end{aligned}$$

Το παρακάτω σύνολο αποτελεί επανεγγραφή ΣΣΕ για τα $\mathcal{Q}, \mathcal{T}$ και έχει υπολογιστεί με τον αλγόριθμο PerfectRef:

$$\mathcal{R}_Q = \{\mathcal{Q}, \mathcal{Q}_1\}, \quad \text{με} \quad \mathcal{Q}_1 = \textsf{Καθηγητής}(x) \rightarrow \mathcal{Q}_A(x)$$

Πιο συγκεκριμένα, το ΣΕ $\mathcal{Q}_1$ παράγεται εφαρμόζοντας το πρώτο αξίωμα του TBox στο άτομο $\textsf{διδάσκει}(x,y)$ του $\mathcal{Q}$, αντικαθιστώντας το με το $\textsf{Καθηγητής}(x)$.

Έστω τώρα ότι το αρχικό ερώτημα επεκτείνεται έτσι ώστε να ανακτήσει όλα τα άτομα που διδάσκουν φοιτητές, δηλαδή, το ερώτημα $\mathcal{Q}$ επεκτείνεται με το άτομο $\alpha = \textsf{Φοιτητής}(y)$ και το νέο ΣΕ είναι το ακόλουθο:

$$\mathcal{Q}' = \textsf{διδάσκει}(x,y) \wedge \textsf{Φοιτητής}(y) \rightarrow \mathcal{Q}_A(x).$$

Και πάλι, χρησιμοποιώντας τον αλγόριθμο PerfectRef, μπορούμε να υπολογίσουμε την επανεγγραφή ΣΣΕ για τα $\mathcal{Q}', \mathcal{T}$:

$$\mathcal{R}'_Q = \{\mathcal{Q}', \mathcal{Q}'_1, \mathcal{Q}, \mathcal{Q}_1\}$$

με $\mathcal{Q}'$, $\mathcal{Q}$ και $\mathcal{Q}_1$ ερωτήματα ίδια με αυτά που έχουν υπολογιστεί προηγουμένως, και

$$\mathcal{Q}'_1 = \textsf{διδάσκει}(x,y) \wedge \textsf{διδάσκει}(z,y) \rightarrow \mathcal{Q}_A(x).$$

Πιο συγκεκριμένα, το $\mathcal{Q}'_1$ παράγεται από το $\mathcal{Q}'$ εφαρμόζοντας το δεύτερο αξίωμα του TBox στο άτομο Φοιτητής(y), αντικαθιστώντας το με το διδάσκει(z, y), με z μια νέα μεταβλητή, ενώ το $\mathcal{Q}$ προκύπτει από το $\mathcal{Q}'_1$ εφαρμόζοντας το βήμα μείωσης στα δύο άτομα του. Αξίζει να παρατηρήσουμε ότι η μεταβλητή y στο ερώτημα $\mathcal{Q}'_1$ είναι δεσμευμένη, εφόσον εμφανίζεται και στα δύο άτομα διδάσκει(x, y) και διδάσκει(z, y), ενώ μετά το βήμα μείωσης γίνεται μη-δεσμευμένη. Συνεπώς, τελικά ο αλγόριθμος μπορεί να παράγει το ερώτημα $\mathcal{Q}_1$ από το $\mathcal{Q}$ όπως δείξαμε προηγουμένως.

Από το παραπάνω παράδειγμα παρατηρούμε πως όταν ο αλγόριθμος PerfectRef εκτελείται για τα $\mathcal{Q}', \mathcal{T}$ χρειάζεται να επαναϋπολογίσει τα ερωτήματα $\mathcal{Q}$ και $\mathcal{Q}_1$, παρόλο που αυτά έχουν ήδη υπολογιστεί για τα $\mathcal{Q}, \mathcal{T}$. Για να αποφύγουμε την επανάληψη αυτού του υπολογισμού θα ήταν ευεργετικό οποιαδήποτε εργασία επανεγγραφής να εφαρμοστεί μόνο για το νεοεισαχθέν άτομο και στη συνέχεια να «συνδυάσουμε» κατάλληλα το αποτέλεσμα με την ήδη υπολογισμένη επανεγγραφή, στην οποία στη συνέχεια θα αναφερόμαστε ως *επανεγγραφή αναφοράς* (*reference rewriting*).

Έστω το ερώτημα $\mathcal{Q}_\alpha =$ Φοιτητής$(y) \rightarrow \mathcal{Q}_A(y)$. Το σύνολο $\mathcal{R}_\alpha = \{\mathcal{Q}_\alpha, \mathcal{Q}'_\alpha\}$, όπου $\mathcal{Q}'_\alpha =$ διδάσκει$(z, y) \rightarrow \mathcal{Q}_A(y)$ για μια νέα μεταβλητή z, είναι μια επανεγγραφή ΣΣΕ για τα $\mathcal{Q}_\alpha, \mathcal{T}$ που μπορεί να υπολογιστεί με τον αλγόριθμο PerfectRef. Είναι εύκολα ορατό ότι το ερώτημα $\mathcal{Q}'$ του $\mathcal{R}'_Q$ μπορεί να παραχθεί προσθέτοντας τα άτομα του $\mathcal{Q}_\alpha$ στο $\mathcal{Q}$ (το οποία έχουν υπολογιστεί προηγουμένως στο $\mathcal{R}_Q$), ενώ το ερώτημα $\mathcal{Q}'_1$ μπορεί επίσης να παραχθεί προσθέτοντας τα άτομα του $\mathcal{Q}'_\alpha$ στο $\mathcal{Q}$. Παρόλα αυτά, πρέπει να σημειωθεί ότι η προσθήκη των ατόμων του $\mathcal{Q}_\alpha$ στο $\mathcal{Q}_1 \in \mathcal{R}_Q$ παράγει το ΣΕ Καθηγητής$(x) \wedge$ Φοιτητής$(y) \rightarrow \mathcal{Q}_A(x)$ που δεν αποτελεί κομμάτι του $\mathcal{R}'_Q$.

Όλα τα παραπάνω υποδηλώνουν ότι δεδομένου ενός ατόμου α και μιας επανεγγραφής $\mathcal{R}_Q$ για τα $\mathcal{Q}, \mathcal{T}$, μια επανεγγραφή για τα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q}), \mathcal{T}$ μπορεί να υπολογιστεί υπολογίζοντας μια επανεγγραφή $\mathcal{R}_\alpha$ για ένα «ειδικό» ερώτημα $\mathcal{Q}_\alpha$ και στη συνέχεια επεκτείνοντας κατάλληλα τα ερωτήματα στο $\mathcal{R}_Q$ με τα άτομα του σώματος των ερωτημάτων από το $\mathcal{R}_\alpha$. Πιο συγκεκριμένα, για τα α, $\mathcal{R}_Q$, $\mathcal{Q}$ και $\mathcal{T}$ ο αλγόριθμος θα υπολογίσει μια επανεγγραφή ΣΣΕ $\mathcal{R}_\alpha$ για το ερώτημα $\mathcal{Q}_\alpha = \alpha \rightarrow \mathcal{Q}_A(\mathsf{var}(\alpha) \cap \mathsf{var}(\mathcal{Q}))$ και το $\mathcal{T}$ και στη συνέχεια θα προσθέσει τα άτομα κάθε ερωτήματος $\mathcal{Q}'_\alpha \in \mathcal{R}_\alpha$ στα άτομα καθε ερωτήματος $\mathcal{Q}' \in \mathcal{R}$ εάν $\mathsf{avar}(\mathcal{Q}'_\alpha) \subseteq \mathsf{var}(\mathcal{Q}')$.

Διαισθητικά η παραπάνω προσέγγιση είναι εφικτή επειδή η διαδικασία της επανεγγραφής στον αλγόριθμο PerfectRef είναι σε πολύ μεγάλο βαθμό «τοπική» σε σχέση με τα άτομα του ερωτήματος. Για παράδειγμα, η εφαρμογή του βήματος αναδιατύπωσης σε κάποιο άτομο ενός ερωτήματος είναι ανεξάρτητη από τα άλλα άτομα του ερωτήματος. Δυστυχώς όμως, το δεύτερο βήμα του αρχικού αλγορίθμου, αυτό της μείωσης, περιλαμβάνει περισσότερα από ένα άτομα του ερωτήματος και για αυτό το επιχείρημα της ανεξαρτησίας σπάει. Μια ευθεία προσέγγιση θα ήταν να χρησιμοποιείται το βήμα της μείωσης μεταξύ των ερωτημάτων από τις δύο επανεγγραφές. Στο τρέχον παράδειγμα, το άτομο διδάσκει(x, y) στο ερώτημα $\mathcal{Q}$ μπορεί να ενοποιηθεί με το άτομο διδάσκει(z, y) στο ερώτημα $\mathcal{Q}'_\alpha$. Το αποτέλεσμα της ενοποίησης αυτής παράγει όντως το ερώτημα $\mathcal{Q}$ του $\mathcal{R}'_Q$. Παρόλα αυτά, η προσέγγιση αυτή εμφανίζει δύο βασικά μειονεκτήματα. Πρώτον, όπως δείξαμε και στην ενότητα 2.5.1 είναι ευρέως γνωστό ότι όσον αφορά στην απόδοση, το βήμα της μείωσης δεν είναι αποδοτικό αφού μπορεί να έχει ως αποτέλεσμα την παραγωγή εκθετικού αριθμού (πιθανώς περιττών) ερωτημά-

των [117, 132, 29]. Κατά δεύτερον, ακόμα και με αυτή την προσέγγιση το ερώτημα $\mathcal{Q}_1 \in \mathcal{R}'_Q$ δε θα μπορούσε να παραχθεί.

Το βήμα μείωσης παρουσιάστηκε στον αλγόριθμο PerfectRef επειδή ένα αξίωμα I μπορεί να είναι εφαρμόσιμο μόνο στη μείωση κάποιου ΣΕ. Στο τρέχον παράδειγμα, το ερώτημα $\mathcal{Q}_1$ παράγεται από το $\mathcal{Q}$ επειδή η μεταβλητή y που ήταν δεσμευμένη στο $\mathcal{Q}'_1$ γίνεται μη-δεσμευμένη μετά το βήμα της μείωσης. Ένα πλεονέκτημα στη δική μας περίπτωση είναι πως γνωρίζουμε ήδη από την επανεγγραφή αναφοράς ότι το $\mathcal{Q}_1$ μπορεί να παραχθεί από το $\mathcal{Q}$. Για το λόγο αυτό, ο αλγόριθμος μας χρειάζεται μόνο να ελέγχει εάν ένα ΣΕ του $\mathcal{R}_\alpha$ μπορεί να «συγχωνευθεί» με το ΣΕ $\mathcal{Q}$ με τέτοιο τρόπο ώστε όλα τα ερωτήματα που έχουν παραχθεί λόγω του $\mathcal{Q}$ στην επανεγγραφή αναφοράς να μπορούν να παραχθούν και πάλι. Εάν ισχύει αυτό, τότε το $\mathcal{Q}$, αλλά και όλα αυτά τα ερωτήματα (στην περίπτωση μας μόνο το $\mathcal{Q}_1$) θα πρέπει να είναι μέρος του αποτελέσματος. Ο αλγόριθμος που παρουσιάζεται στην επόμενη ενότητα αναγνωρίζει τέτοιες περιπτώσεις χρησιμοποιώντας τη συνάρτηση mergeCQs που ορίζεται στη συνέχεια.

Ορισμός 4.1.1. Έστω $\mathcal{Q}, \mathcal{Q}'$ δύο ΣΕ. Τότε, η συνάρτηση $\mathsf{mergeCQs}(\mathcal{Q}', \mathcal{Q})$ επιστρέφει μια αντικατάσταση Σ που ορίζεται ως εξής:

- Εάν υπάρχει $\alpha \in \mathsf{bd}(\mathcal{Q}) \cap \mathsf{bd}(\mathcal{Q}')$, τότε η Σ περιέχει την αντικατάσταση $\{x \mapsto x \mid x \in \mathsf{var}(\alpha)\}$,
- Εάν υπάρχει $R(z, y) \in \mathcal{Q}', R(x, y) \in \mathcal{Q}$ ή $R(y, z) \in \mathcal{Q}', R(y, x) \in \mathcal{Q}$ και οι μεταβλητές x, y, z είναι ανά δύο διαφορετικές, τότε το Σ περιέχει την αντικατάσταση $\{z \mapsto x\}$.

$\triangle$

Με τη χρήση της συνάρτησης mergeCQs ο αλγόριθμος PerfectRef μπορεί να επεκταθεί στον PerfectRef$^+$ που δίνεται από τον Αλγόριθμο 8, ο οποίος χρησιμοποιεί ένα *περιορισμένο βήμα μείωσης* (*reduced reduction step*), ελαχιστοποιώντας τις περιπτώσεις παραγωγής μειωμένων ερωτημάτων. Αξίζει να σημειωθεί πως το περιορισμένο βήμα μείωσης είναι παρόμοιο με το *βήμα παραγοντοποίησης* (*factorization step*) που προτάθηκε από τους Gottlob et al. [64]. Παρόλα αυτά, εφόσον η DL-Lite επιτρέπει την ύπαρξη κατηγορημάτων που περιέχουν το πολύ δύο μεταβλητές οι έλεγχοι που χρειάζεται να πραγματοποιήσουμε (αυτοί που δίνονται από τον Ορισμό 4.1.1) είναι πιο στοχευμένοι και απαιτούν μικρότερο υπολογιστικό κόστος. Από την άλλη, για την υλοποίηση του βήματος παραγοντοποίησης είναι απαραίτητο να γίνει έλεγχος για το εάν αυτό είναι δυνατό να εφαρμοστεί για συγκεκριμένα αξιώματα του TBox όσον αφορά κάποιο άτομο, κάτι το οποίο είναι (σχεδόν) ισοδύναμο με την εφαρμογή ενός βήματος επανεγγραφής.

Όπως δείχνει το επόμενο παράδειγμα, η αντικατάσταση που επιστρέφεται από τη συνάρτηση mergeCQs υποδεικνύει τον τρόπο με τον οποίο ένα ερώτημα μπορεί να συγχωνευθεί με ένα άλλο και αποτελεί καίριο σημείο για τον αλγόριθμο μας.

Παράδειγμα 4.1.2. Έστω το παρακάτω TBox και ΣΕ:

$$\mathcal{T} = \{A(x) \to \exists y.R(x, y)\} \quad \mathcal{Q} = R(x, y) \to \mathcal{Q}_A(x).$$

Αλγόριθμος 8 $\mathsf{PerfectRef}^{+}(\mathcal{Q}, \mathcal{T})$

Είσοδος: Ένα ΣΕ $\mathcal{Q}$ και ένα DL-Lite$_R$-TBox $\mathcal{T}$

1: Αρχικοποίησε μια επανεγγραφή ΣΣΕ $\mathcal{R}'_Q := \{\mathcal{Q}\}$
2: **repeat**
3: $\quad\mathcal{R}_Q := \mathcal{R}'_Q$
4: $\quad$**for all** $\mathcal{Q}_1 \in \mathcal{R}_Q$ **do**
5: $\quad\quad$**for all** $\alpha \in \mathcal{Q}_1$ **do**
6: $\quad\quad\quad$**for all** $I \in \mathcal{T}$ **do**
7: $\quad\quad\quad\quad$**if** I είναι εφαρμόσιμο στο α **then**
8: $\quad\quad\quad\quad\quad$Πρόσθεσε το $\mathcal{Q}_2 := \mathcal{Q}_1[\alpha/\mathsf{gr}(\alpha, I)]$ στο $\mathcal{R}'_Q$
9: $\quad\quad\quad\quad$**end if**
10: $\quad\quad\quad$**end for**
11: $\quad\quad$**end for**
12: $\quad\quad$**for all** α_1, α_2 στο $\mathcal{Q}_1$ **do**
13: $\quad\quad\quad\sigma = \mathsf{mergeCQs}(\alpha_1 \to \mathsf{hd}(\mathcal{Q}_1), \alpha_2 \to \mathsf{hd}(\mathcal{Q}_1))$
14: $\quad\quad\quad$**if** $\sigma \neq \emptyset$ **then**
15: $\quad\quad\quad\quad$Πρόσθεσε το $\mathcal{Q}_1\sigma$ στο $\mathcal{R}'_Q$
16: $\quad\quad\quad$**end if**
17: $\quad\quad$**end for**
18: $\quad$**end for**
19: **until** $\mathcal{R}_Q = \mathcal{R}'_Q$
20: **return** $\mathcal{R}_Q$

που έχουν την παρακάτω επανεγγραφή ΣΣΕ:

$$\mathcal{R}_Q = \{\mathcal{Q}, \mathcal{Q}_1\}, \quad \text{με} \quad \mathcal{Q}_1 = A(x) \to \mathcal{Q}_A(x)$$

Έστω τώρα ότι επεκτείνουμε το ΣΕ $\mathcal{Q}$ με το άτομο $\alpha_1 = R(z, y)$ και ότι θέλουμε να υπολογίσουμε μια επανεγγραφή ΣΣΕ για τα $\mathcal{Q}' = \bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{R(z, y)\} \to \mathsf{hd}(\mathcal{Q})$ και $\mathcal{T}$, χρησιμοποιώντας την προσέγγιση που περιγράψαμε προηγουμένως. Πρώτα, κατασκευάζουμε το ερώτημα $\mathcal{Q}_{\alpha_1} = R(z, y) \to \mathcal{Q}_A(y)$ και στη συνέχεια την επανεγγραφή ΣΣΕ $\mathcal{R}_{\alpha_1} = \{\mathcal{Q}_{\alpha_1}\}$ για τα $\mathcal{Q}_{\alpha_1}, \mathcal{T}$. Τέλος, ένα ΣΣΕ $\mathcal{R}'_Q$ κατασκευάζεται όπως φαίνεται παρακάτω: Τα άτομα του $\mathcal{Q}_{\alpha_1}$ προστίθενται στο $\mathcal{Q}$ κατασκευάζοντας το ΣΕ $\mathcal{Q}' = \bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{R(z, y)\} \to \mathsf{hd}(\mathcal{Q})$, ενώ το ερώτημα $\mathcal{Q}_{\alpha_1}$ συγχωνεύεται με το $\mathcal{Q}$ με $\mathsf{mergeCQs}(\mathcal{Q}_{\alpha_1}, \mathcal{Q}) = \{z \mapsto x\}$, συνεπώς τα ερωτήματα $\mathcal{Q}$ και $\mathcal{Q}_1$ προστίθενται στο $\mathcal{R}'_Q$. Μπορεί να επαληθευτεί ότι το $\mathcal{R}'_Q = \{\mathcal{Q}', \mathcal{Q}, \mathcal{Q}_1\}$ είναι μια επανεγγραφή ΣΣΕ για τα $\mathcal{Q}', \mathcal{T}$.

Έστω τώρα ότι θέλουμε να επεκτείνουμε περαιτέρω το $\mathcal{Q}'$ με το άτομο $\alpha_2 = B(z)$ και ότι θέλουμε να υπολογίσουμε μια επανεγγραφή ΣΣΕ για τα $\mathcal{Q}'' = \bigwedge \mathsf{bd}(\mathcal{Q}') \cup \{B(z)\} \to \mathsf{hd}(\mathcal{Q}')$ και $\mathcal{T}$. Μια τέτοια επανεγγραφή, που έχει υπολογιστεί με τον αλγόριθμο PerfectRef είναι το ακόλουθο σύνολο:

$$\begin{aligned} \mathcal{R}''_Q = \{\mathcal{Q}'', \mathcal{Q}'_1, \mathcal{Q}'_2\}, \quad \text{με} \quad & \mathcal{Q}'_1 = R(x, y) \wedge B(x) \to \mathcal{Q}_A(x) \\ \text{και} \quad & \mathcal{Q}'_2 = A(x) \wedge B(x) \to \mathcal{Q}_A(x) \end{aligned}$$

Έστω τώρα ότι θέλουμε να υπολογίσουμε το $\mathcal{R}''_Q$ χρησιμοποιώντας την προσέγγιση μας. Για το λόγο αυτό, κατασκευάζουμε το ερώτημα $\mathcal{Q}_{\alpha_2} := B(z) \rightarrow \mathcal{Q}_A(z)$, την επανεγγραφή ΣΣΕ $\mathcal{R}_{\alpha_2} := \{\mathcal{Q}_{\alpha_2}\}$ για τα $\mathcal{Q}_{\alpha_2}, \mathcal{T}$ και στη συνέχεια συνδυάζουμε τα ερωτήματα από το $\mathcal{R}'_Q$ και το $\mathcal{R}_{\alpha_2}$. Προσθέτοντας τα άτομα του $\mathcal{Q}_{\alpha_2}$ στο $\mathcal{Q}'$ κατασκευάζεται το ερώτημα $\mathcal{Q}''$, ενώ, για όλα τα άλλα ερωτήματα $\mathcal{Q}_i \in \mathcal{R}'_Q$ έχουμε ότι $\mathsf{avar}(\mathcal{Q}_{\alpha_2}) \nsubseteq \mathsf{var}(\mathcal{Q}_i)$ και συνεπώς κανένα άλλο ερώτημα του $\mathcal{R}''_Q$ δεν μπορεί να παραχθεί. ◇

Το πρόβλημα στο προηγούμενο παράδειγμα είναι ότι μετά την συγχώνευση του $\mathcal{Q}_{\alpha_1}$ με το $\mathcal{Q}$ η αναφορά μας στη μεταβλητή z χάθηκε καθώς αυτή αντικαταστάθηκε από την μεταβλητή x. Η πληροφορία αυτή είναι ιδιαίτερα σημαντική στις περιπτώσεις που θέλουμε να επεκτείνουμε περαιτέρω ένα ερώτημα και να αποφασίσουμε εάν είναι εφικτό να επεκταθεί ένα ερώτημα από το $\mathcal{R}'_Q$ με άτομα από τα ερωτήματα του $\mathcal{R}_{\alpha_2}$. Για να διαχειριστούμε σωστά αυτές τις περιπτώσεις, αντί του ελέγχου υποσυνόλου ανάμεσα στις μεταβλητές, ο αλγόριθμος μας χρησιμοποιεί την πιο σύνθετη μέθοδο canBeJoined που ορίζεται στη συνέχεια.

Ορισμός 4.1.3. Έστω $\mathcal{Q}$ ΣΕ, έστω σ αντικατάσταση, και έστω vars ένα σύνολο από μεταβλητές. Η συνάρτηση $\mathsf{canBeJoined}(\mathcal{Q}, \sigma, \mathsf{vars})$ επιστρέφει αληθές εάν για κάθε $z \in \mathsf{vars}$ υπάρχει $x \in \mathsf{var}(\mathcal{Q})$ τέτοιο ώστε για τον γράφο G που επάγεται από την σ, έχουμε $z \rightsquigarrow_G x$. △

Παράδειγμα 4.1.4. Έστω το ερώτημα $\mathcal{Q}''$, τα ΣΣΕ $\mathcal{R}_{\alpha_2}$ και $\mathcal{R}'_Q$, και η αντικατάσταση σ από το Παράδειγμα 4.1.2. Για τα ερωτήματα $\mathcal{Q}_{\alpha_2} \in \mathcal{R}_{\alpha_2}$ και $\mathcal{Q} \in \mathcal{R}'_Q$ έχουμε ότι $\mathsf{canBeJoined}(\mathcal{Q}, \sigma, \mathsf{avar}(\mathcal{Q}_{\alpha_2})) =$ αληθές. Συνεπώς, τα άτομα του $\mathcal{Q}_{\alpha_2}$ (δηλαδή το $B(z)$) μπορούν να προστεθούν σε αυτά του $\mathcal{Q}$. Πρέπει ωστόσο να σημειωθεί ότι λόγω της σ το νέο ερώτημα που πρέπει να δημιουργηθεί είναι το ερώτημα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{B(z)\sigma\} \rightarrow \mathsf{hd}(\mathcal{Q})$, που κατ' ακρίβεια είναι το $\mathcal{Q}'_1$ από το Παράδειγμα 4.1.2. Όμοια για το ερώτημα $\mathcal{Q}_1 \in \mathcal{R}'_Q$ έχουμε $\mathsf{canBeJoined}(\mathcal{Q}_1, \sigma, \mathsf{avar}(\mathcal{Q}_{\alpha_2})) =$ αληθές, συνεπώς κατασκευάζεται το ερώτημα $\bigwedge \mathsf{bd}(\mathcal{Q}_1) \cup \{B(z)\sigma\} \rightarrow \mathsf{hd}(\mathcal{Q}_1)$ (δηλαδή το $\mathcal{Q}'_2$). Συνεπώς η επανεγγραφή ΣΣΕ $\mathcal{R}''_Q$ για τα $\mathcal{Q}'', \mathcal{T}$ μπορεί να κατασκευαστεί. ◇

Συνοψίζοντας, όταν ένα ερώτημα $\mathcal{Q}_i \in \mathcal{R}_\alpha$ συγχωνεύεται με κάποιο ερώτημα $\mathcal{Q}_j \in \mathcal{R}_Q$ πρέπει να γνωρίζουμε πρώτα από όλα ποιά ερωτήματα έχουν παραχθεί στο $\mathcal{R}_Q$ εξαιτίας του $\mathcal{Q}_j$, έτσι ώστε να τα προσθέσουμε στο αποτέλεσμα, και κατά δεύτερον, ποιές αντικαταστάσεις μεταβλητών χρησιμοποιούνται στη συγχώνευση. Για να μπορέσουμε να συλλάβουμε την πληροφορία αυτή, σε αντίθεση με άλλες προσεγγίσεις που χρησιμοποιούν τις επανεγγραφές ΣΣΕ $\mathcal{R}_Q$ και $\mathcal{R}_\alpha$, ο αλγόριθμος μας λειτουργεί πάνω σε γράφους ερωτημάτων $\mathcal{G}$ και $\mathcal{G}_\alpha$ οι οποίοι αποθηκεύουν τις συσχετίσεις μεταξύ των ερωτημάτων καθώς και τις παραγόμενες αντικαταστάσεις.

Ορισμός 4.1.5. Έστω $\mathcal{Q}$ ΣΕ και έστω $\mathcal{T}$ DL-Lite$_R$-TBox. Ένας *γράφος επανεγγραφής* για τα $\mathcal{Q}, \mathcal{T}$ είναι ένας κατευθυνόμενος γράφος $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$, με $\mathcal{R}_Q$ μια επανεγγραφή ΣΣΕ για τα $\mathcal{Q}, \mathcal{T}$, $\mathcal{H}$ μια δυαδική σχέση πάνω στην $\mathcal{R}_Q$, και m αντιστοίχιση του κάθε $\mathcal{Q}_i \in \mathcal{R}_Q$ σε ένα σύνολο μεταβλητών. Επιπλέον, ο γράφος $\mathcal{G}$ ικανοποιεί τις παρακάτω ιδιότητες:

- Εάν $\langle \mathcal{Q}_1, \mathcal{Q}_2 \rangle \in \mathcal{H}$, τότε το ερώτημα $\mathcal{Q}_2$ παράγεται από το $\mathcal{Q}_1$ με την εφαρμογή ενός βήματος αναδιατύπωσης ή μείωσης.
- Για κάθε $\langle \mathcal{Q}_1, \mathcal{Q}_2 \rangle \in \mathcal{H}$ εάν το $\mathcal{Q}_2$ παράγεται από ένα βήμα αναδιατύπωσης, τότε ισχύει $m(\mathcal{Q}_2) = m(\mathcal{Q}_1)$, ενώ εάν παράγεται από ένα βήμα μείωσης με τον πιο γενικό ενοποιητή σ, τότε $m(\mathcal{Q}_2) = m(\mathcal{Q}_1) \cup \sigma$.

$\triangle$

Ένα πολύ σημαντικό ερώτημα που εγείρεται στο σημείο αυτό είναι εάν μπορούμε να υπολογίσουμε την επανεγγραφή ΣΣΕ δεδομένης οποιαδήποτε επανεγγραφής αναφοράς. Όπως φαίνεται και από το παρακάτω παράδειγμα, κάτι τέτοιο δεν είναι πάντα εφικτό.

Παράδειγμα 4.1.6. Θεωρούμε τα παρακάτω TBox και ΣΕ:

$$\mathcal{T} = \{A(x) \rightarrow \exists y.R(x,y)\} \quad \mathcal{Q} = A(x) \wedge R(x,y) \rightarrow \mathcal{Q}_A(x).$$

Τότε, $\mathcal{G}_1 = \langle \mathcal{R}_1, \mathcal{H}, \emptyset \rangle$ με $\mathcal{R}_1 = \{\mathcal{Q}, \mathcal{Q}_1\}$, $\mathcal{Q}_1 = A(x) \rightarrow \mathcal{Q}_A(x)$ και $\mathcal{H} = \{\langle \mathcal{Q}, \mathcal{Q}_1 \rangle\}$ είναι ένας γράφος επανεγγραφής για τα $\mathcal{Q}, \mathcal{T}$. Όμως, το $\mathcal{Q}$ είναι περιττό στο $\mathcal{R}_1$. Συνεπώς, το $\mathcal{R}_2 = \{\mathcal{Q}_1\}$ είναι επίσης επανεγγραφή ΣΣΕ για τα $\mathcal{Q}, \mathcal{T}$ και ο $\mathcal{G}_2 = \langle \mathcal{R}_2, \emptyset, \emptyset \rangle$ είναι γράφος επανεγγραφής για τα $\mathcal{Q}, \mathcal{T}$.

Έστω τώρα ότι θέλουμε να επεκτείνουμε το $\mathcal{Q}$ με το άτομο $\alpha = B(y)$ κατασκευάζοντας το ερώτημα $\mathcal{Q}' = \bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{B(y)\} \rightarrow \mathsf{hd}(\mathcal{Q})$. Μια επανεγγραφή ΣΣΕ για τα $\mathcal{Q}', \mathcal{T}$ υπολογισμένη με τη χρήση του αλγορίθμου PerfectRef αποτελείται από το ΣΣΕ $\mathcal{R}'_Q = \{\mathcal{Q}'\}$.

Θεωρούμε τώρα το ερώτημα $\mathcal{Q}_\alpha = B(y) \rightarrow \mathcal{Q}_A(y)$. Με χρήση του PerfectRef υπολογίζουμε την επανεγγραφή ΣΣΕ $\mathcal{R}_\alpha = \{\mathcal{Q}_\alpha\}$. Όπως είναι εμφανές δεν είναι δυνατό να υπολογίσουμε το $\mathcal{R}'_Q$ χρησιμοποιώντας την επανεγγραφή ΣΣΕ $\mathcal{R}_2$ που περιέχει μη-περιττά ερωτήματα. Το $\mathcal{R}'_Q$ μπορεί να κατασκευαστεί με τη χρήση της επανεγγραφής ΣΣΕ $\mathcal{R}_1$ που περιέχει περιττά ερωτήματα. Πιο συγκεκριμένα, η προσθήκη των ατόμων του $\mathcal{Q}_\alpha$ στο $\mathcal{Q}$ έχει ως αποτέλεσμα τη δημιουργία του ΣΕ $\mathcal{Q}'$. $\diamondsuit$

Το πρόβλημα στο προηγούμενο παράδειγμα έγκειται στο ότι παρόλο που το ερώτημα $\mathcal{Q}$ είναι περιττό στο $\mathcal{R}_1$ τα ερωτήματα που παράγονται από αυτό δεν είναι περιττά στο τελικό ΣΣΕ και συνεπώς το ερώτημα αυτό είναι «σχετικό» για τον υπολογισμό του ΣΣΕ για το $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q})$. Πιο συγκεκριμένα, το $\mathcal{Q}_1$ που χρησιμοποιείται για να αφαιρέσουμε το $\mathcal{Q}$, δεν παράγει κανένα ΣΕ που να αποτελεί κομμάτι της τελικής επανεγγραφής ΣΣΕ. Στη συνέχεια παρουσιάζουμε με τυπικό τρόπο μια ιδιότητα για την επανεγγραφή αναφοράς που χρησιμοποιείται σαν είσοδος στον αγλόριθμο μας. Η ιδιότητα αυτή είναι ικανή για την παραγωγή ενός γράφου επανεγγραφής για ένα επεκτεταμένο ερώτημα.

Ορισμός 4.1.7. Έστω $\mathcal{Q}$ ένα ερώτημα, έστω $\mathcal{T}$ ένα DL-Lite$_R$-TBox, και έστω $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$ ένας γράφος επανεγγραφής για τα $\mathcal{Q}, \mathcal{T}$. Λέμε ότι ο $\mathcal{G}$ είναι *κλειστός ως προς την αναδιατύπωση* (*reformulation closed*) για τα $\mathcal{Q}, \mathcal{T}$ εάν ικανοποιούνται οι παρακάτω ιδιότητες:

1. Το $\mathcal{Q}$ είναι κόμβος κορυφής στον $\mathcal{G}$.
2. Για κάθε κόμβο κορυφής $\mathcal{Q}_i$ στον $\mathcal{G}$ έχουμε ότι $m(\mathcal{Q}_i) = \emptyset$.
3. Εάν ένα ερώτημα $\mathcal{Q}_2$ μπορεί να παραχθεί χρησιμοποιώντας ένα απλό βήμα αναδιατύπωσης σε κάποιο ερώτημα $\mathcal{Q}_1 \in u$, τότε $\langle \mathcal{Q}_1, \mathcal{Q}_2 \rangle \in \mathcal{H}$.
4. Εάν $\mathcal{Q}_1 \in \mathcal{R}_Q$ και τα άτομα $R(z, y)$ και $R(x, y)$ ή τα άτομα $R(y, z)$ και $R(y, x)$ εμφανίζονται στο $\mathcal{Q}_1$, τότε $\langle \mathcal{Q}_1, \mathcal{Q}_2 \rangle \in \mathcal{H}$, με $\mathcal{Q}_2 = \mathcal{Q}_1\{z \mapsto x\}$.

$\triangle$

Παρατηρούμε ότι ο γράφος επανεγγραφής $\mathcal{G}_2$ από το Παράδειγμα 4.1.6 δεν είναι κλειστός ως προς την αναδιατύπωση για τα $\mathcal{Q}, \mathcal{T}$ καθώς το $\mathcal{Q}$ δεν εμφανίζεται ως κόμβος κορυφής. Ο $\mathcal{G}_1$ όμως, είναι κλειστός ως προς την αναδιατύπωση για τα $\mathcal{Q}, \mathcal{T}$.

4.2 Αλγόριθμος Επέκτασης Ερωτημάτων

Αλγόριθμος 9 ExtendRewritingForNewAtom$(\mathcal{T}, \mathcal{G}, \alpha)$

Είσοδος: Ένα TBox $\mathcal{T}$, ένας γράφος επανεγγραφής $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$ για κάποιο ΣΕ $\mathcal{Q}$ μ.β.τ. $\mathcal{T}$ και ένα νέο άτομο α.

1: $\mathcal{G}_\alpha :=$ ex-PerfectRef$(\alpha \rightarrow \mathcal{Q}_A(\mathsf{var}(\alpha) \cap \mathsf{var}(\mathcal{Q})), \mathcal{T})$
2: $\mathcal{G}' :=$ joinGraphs$(\mathcal{G}, \mathcal{G}_\alpha, \mathsf{var}(\alpha) \cap \mathsf{var}(\mathcal{Q}), \mathsf{avar}(\mathcal{Q}))$
3: **return** $\mathcal{G}'$

Ο αλγόριθμος μας για τον υπολογισμό ενός γράφου επανεγγραφής για ένα ερώτημα $\mathcal{Q}$ που επεκτείνεται με ένα άτομο α φαίνεται στον Αλγόριθμο 9. Ο αλγόριθμος δέχεται σαν είσοδο ένα TBox $\mathcal{T}$, ένα γράφο επανεγγραφής $\mathcal{G}$ για τα $\mathcal{Q}, \mathcal{T}$, και ένα άτομο α και επιστρέφει έναν γράφο επανεγγραφής για τα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q}), \mathcal{T}$. Ο αλγόριθμος υπολογίζει πρώτα έναν γράφο επανεγγραφής $\mathcal{G}_\alpha$ για το ερώτημα $\alpha \rightarrow \mathcal{Q}_A(\mathsf{var}(\alpha) \cap \mathsf{var}(\mathcal{Q}))$, ο οποίος εξάγει τη γνώση του $\mathcal{T}$ που αφορά το άτομο α. Αυτό πραγματοποιείται με τη χρήση της υπο-ρουτίνας ex-PerfectRef, που φαίνεται στον Αλγόριθμο 10. Ο αλγόριθμος αυτός είναι μια ελαφρώς πιο εκτεταμένη έκδοση του αλγόριθμου PerfectRef [31], που εκτός από την εφαρμογή των βημάτων αναδιατύπωσης και μείωσης, αποθηκεύει επίσης τις εξαρτήσεις παραγωγής ανάμεσα στα ερωτήματα, με τη χρήση ενός γράφου. Στη συνέχεια, ο Αλγόριθμος 9 «συνδυάζει» τους $\mathcal{G}$ και $\mathcal{G}_\alpha$ χρησιμοποιώντας τον αλγόριθμο joinGraphs και υπολογίζει έναν γράφο επανεγγραφής για το επεκτεταμένο ερώτημα.

Ο αλγόριθμος joinGraphs φαίνεται στον Αλγόριθμο 11. Διαισθητικά, ο αλγόριθμος αυτός υπολογίζει το Καρτεσιανό γινόμενο των δύο γράφων επανεγγραφής που παίρνει ως είσοδο. Η διαίσθηση είναι ότι εάν $\langle \mathcal{Q}, \mathcal{Q}' \rangle \in \mathcal{G}$ (ήτοι, το $\mathcal{Q}'$ παράγεται από το $\mathcal{Q}$) και το $\mathcal{Q}_\alpha$ είναι ένας κόμβος στον γράφο $\mathcal{G}_\alpha$, τότε το ίδιο βήμα θα είναι επίσης εφαρμόσιμο στο ερώτημα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \mathsf{bd}(\mathcal{Q}_\alpha) \rightarrow \mathsf{hd}(\mathcal{Q})$—δηλαδή, το ερώτημα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \mathsf{bd}(\mathcal{Q}_\alpha) \rightarrow \mathsf{hd}(\mathcal{Q})$ θα παράγει το ΣΕ $\bigwedge \mathsf{bd}(\mathcal{Q}') \cup \mathsf{bd}(\mathcal{Q}_\alpha) \rightarrow \mathsf{hd}(\mathcal{Q}')$. Το ίδιο θα ισχύει και για το $\mathcal{Q}$ που είναι κόμβος στον $\mathcal{G}$ και τη σχέση $\langle \mathcal{Q}_\alpha, \mathcal{Q}'_\alpha \rangle \in \mathcal{G}_\alpha$.

Αλγόριθμος 10 ex-PerfectRef($\mathcal{Q}, \mathcal{T}$)

Είσοδος: Ένα ΣΕ $\mathcal{Q}$ και ένα DL-Lite$_R$-TBox $\mathcal{T}$

1: Αρχικοποίησε μια επανεγγραφή ΣΣΕ $\mathcal{R}_Q := \{\mathcal{Q}\}$
2: Αρχικοποίησε μια αντιστοίχιση m από ΣΕ σε αντικαταστάσεις
3: $\mathcal{G} := \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$
4: $m(\mathcal{Q}) := \emptyset$
5: **repeat**
6: $\quad \mathcal{R}'_Q := \mathcal{R}_Q$
7: $\quad$ **for all** $\mathcal{Q}_1 \in \mathcal{R}'_Q$ **do**
8: $\quad\quad$ **for all** $\alpha \in \mathcal{Q}_1$ **do**
9: $\quad\quad\quad$ **for all** $I \in \mathcal{T}$ **do**
10: $\quad\quad\quad\quad$ **if** I είναι εφαρμόσιμο στο α **then**
11: $\quad\quad\quad\quad\quad$ Πρόσθεσε το $\mathcal{Q}_2 := \mathcal{Q}_1[\alpha/\mathsf{gr}(\alpha, I)]$ στο $\mathcal{R}'_Q$
12: $\quad\quad\quad\quad\quad$ Πρόσθεσε το $\langle \mathcal{Q}_1, \mathcal{Q}_2 \rangle$ στο $\mathcal{G}$
13: $\quad\quad\quad\quad\quad$ $m(\mathcal{Q}_2) := m(\mathcal{Q}_1)$
14: $\quad\quad\quad\quad$ **end if**
15: $\quad\quad\quad$ **end for**
16: $\quad\quad$ **end for**
17: $\quad\quad$ **for all** α_1, α_2 στο $\mathcal{Q}_1$ **do**
18: $\quad\quad\quad$ **if** υπάρχει mgu σ για τα α_1 και α_2 **then**
19: $\quad\quad\quad\quad$ Πρόσθεσε το $\langle \mathcal{Q}_1, \mathcal{Q}_1\sigma \rangle$ στο $\mathcal{G}$
20: $\quad\quad\quad\quad$ $m(\mathcal{Q}_1\sigma) := m(\mathcal{Q}_1) \cup \{\sigma\}$
21: $\quad\quad\quad$ **end if**
22: $\quad\quad$ **end for**
23: $\quad$ **end for**
24: **until** $\mathcal{R}_Q = \mathcal{R}'_Q$
25: **return** $\mathcal{G}$

Πιο συγκεκριμένα, ο αλγόριθμος χρησιμοποιεί τις ουρές Q και Q_α έτσι ώστε να προσπελάσει τους γράφους επανεγγραφής $\mathcal{G}$ και $\mathcal{G}_\alpha$, αντίστοιχα. Στη γραμμή 5 επιλέγει ένα στοιχείο $\mathcal{Q}_h$ από τον $\mathcal{G}$ και ελέγχει εάν το ερώτημα αυτό μπορεί να επεκταθεί με τα άτομα των ερωτημάτων του $\mathcal{G}_\alpha$ (γραμμή 6). Εάν μπορεί, τότε επιλέγεται επίσης ένα ΣΕ $\mathcal{Q}_\alpha$ από τον $\mathcal{G}_\alpha$ (γραμμή 9), παράγεται ένα νέο ερώτημα $\mathcal{Q}_c$ από τα $\mathcal{Q}_h$ και $\mathcal{Q}_\alpha$ (γραμμή 11) διακεκριμένες μεταβλητές το nv. Επιπλέον, η $m(\mathcal{Q}_h)$ ορίζεται ως η αντικατάσταση για το $\mathcal{Q}_c$ (γραμμή 12). Στη συνέχεια, ο αλγόριθμος ελέγχει εάν το $\mathcal{Q}_\alpha$ μπορεί να συγχωνευθεί με το $\mathcal{Q}_h$. Εάν κάτι τέτοιο μπορεί να γίνει, τότε προσθέτει την $\langle \mathcal{Q}_c, \mathcal{Q}_h\sigma \rangle$ (όπου το $\mathcal{Q}_h\sigma$ είναι το ερώτημα που προκύπτει από την εφαρμογή της αντικατάστασης σ στο $\mathcal{Q}_h$) στον νέο γράφο (γραμμή 15) και στη συνέχεια «αντιγράφει» στον $\mathcal{G}'$ το κομμάτι του $\mathcal{G}$ που έχει παραχθεί εξαιτίας του $\mathcal{Q}_h$ (γραμμές 16–25) εφαρμόζοντας παράλληλα την αντικατάσταση μ' που υπολογίζεται με τη βοήθεια της συνάρτησης buildSubst.

Ορισμός 4.2.1. Έστω σ και κ σύνολα από αντιστοιχήσεις. Τότε η συνάρτηση buildSubst(σ, κ) επιστρέφει ένα νέο σύνολο από αντιστοιχήσεις μ το οποίο κατασκευά-

Αλγόριθμος 11 joinGraphs$(\mathcal{G}, \mathcal{G}_\alpha, jv, av)$

Είσοδος: Ένας γράφος επανεγγραφής $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$ για κάποια $\mathcal{Q}, \mathcal{T}$ και ένας γράφος επανεγγραφής $\mathcal{G}_\alpha$, το σύνολο jv των σημείων-ένωσης, και ένα σύνολο μεταβλητών av.

1: Αρχικοποίησε ένα ΣΣΕ $\mathcal{R}'_Q := \emptyset$ και μια δυαδική σχέση $\mathcal{H}' := \emptyset$ και μια αντιστοίχιση $m' = \emptyset$
2: $\mathcal{G}' := \langle \mathcal{R}'_Q, \mathcal{H}', m' \rangle$
3: Αρχικοποίησε μια ουρά Q με το ΣΕ $\mathcal{Q}$
4: **while** $Q \neq \emptyset$ **do**
5: Αφαίρεσε την κεφαλή $\mathcal{Q}_h$ της Q και όρισε $\kappa := m(\mathcal{Q}_h)$
6: **if** canBeJoined$(\mathcal{Q}_h, \kappa, jv)$ **then**
7: Αρχικοποίησε μια ουρά Q_α με ένα κόμβο κορυφής του $\mathcal{G}_\alpha$
8: **while** $Q_\alpha \neq \emptyset$ **do**
9: Αφαίρεσε την κεφαλή $\mathcal{Q}_\alpha$ της Q_α
10: $nv :=$ avar$(\mathcal{Q}_h) \cup$ (var$(\mathcal{Q}_\alpha \kappa) \cap av)$
11: $\mathcal{Q}_c := \bigwedge$ bd$(\mathcal{Q}_h) \cup$ bd$(\mathcal{Q}_\alpha)\kappa \rightarrow \mathcal{Q}_A(nv)$
12: $m'(\mathcal{Q}_c) := \kappa$
13: Πρόσθεσε το $\mathcal{Q}_c$ στο $\mathcal{R}'_Q$
14: **for all** $\sigma \in$ mergeCQs$(\mathcal{Q}_\alpha \kappa, \mathcal{Q}_h)$ **do**
15: Προσέθεσε την $\langle \mathcal{Q}_c, \mathcal{Q}_h \sigma \rangle$ to $\mathcal{G}'$
16: **for all** $\mathcal{Q}'$ s.t. $\mathcal{Q}_h \rightsquigarrow_{\mathcal{G}} \mathcal{Q}'$ **do**
17: **if** dom$(\sigma) \subseteq$ var$(\mathcal{Q}') \cup$ dom$(m(\mathcal{Q}'))$ **then**
18: $\mu' :=$ buildSubst$(\sigma, m(\mathcal{Q}'))$
19: $m'([\mathcal{Q}' \rightarrow \mathcal{Q}_A(nv)]\mu') := \mu'$
20: **for all** $\langle \mathcal{Q}', \mathcal{Q}'' \rangle \in \mathcal{G}$ **do**
21: $\mu'' :=$ buildSubst$(\sigma, m(\mathcal{Q}''))$
22: Προσέθεσε την $\langle [\mathcal{Q}' \rightarrow \mathcal{Q}_A(nv)]\mu', [\mathcal{Q}'' \rightarrow \mathcal{Q}_A(nv)]\mu'' \rangle$ στον $\mathcal{G}'$
23: **end for**
24: **end if**
25: **end for**
26: **end for**
27: **for all** $\langle \mathcal{Q}_h, \mathcal{Q}' \rangle \in \mathcal{G}$ **do**
28: buildChildren$(\mathcal{Q}_c, \mathcal{Q}', \mathcal{Q}_\alpha, \mathcal{G}, \mathcal{G}', av, jv)$
29: Πρόσθεσε το $\mathcal{Q}'$ στην Q
30: **end for**
31: **for all** $\langle \mathcal{Q}_\alpha, \mathcal{Q}' \rangle \in \mathcal{G}_\alpha$ **do**
32: buildChildren$(\mathcal{Q}_c, \mathcal{Q}_h, \mathcal{Q}', \mathcal{G}, \mathcal{G}', av, jv)$
33: Πρόσθεσε το $\mathcal{Q}'$ στην Q_α
34: **end for**
35: **end while**
36: **end if**
37: **end while**
38: **return** $\mathcal{G}'$

Αλγόριθμος 12 buildChildren$(\mathcal{Q}_c, \mathcal{Q}, \mathcal{Q}_\alpha, \mathcal{G}, \mathcal{G}', av, jv)$

Είσοδος: $\mathcal{Q}_c$, $\mathcal{Q}$, $\mathcal{Q}_\alpha$ ΣΕ, $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$ και $\mathcal{G}' = \langle \mathcal{R}'_Q, \mathcal{H}', m' \rangle$ γράφοι και av, jv σύνολα από μεταβλητές.

1: $\kappa := m(\mathcal{Q})$
2: **if** canBeJoined$(\mathcal{Q}, \kappa, jv)$ **then**
3: $\quad nv :=$ avar$(\mathcal{Q}) \cup$ (var$(\mathcal{Q}_\alpha\kappa) \cap av)$
4: $\quad \mathcal{Q}_n := \bigwedge$ bd$(\mathcal{Q}) \cup$ bd$(\mathcal{Q}_\alpha)\kappa \rightarrow \mathcal{Q}_A(nv)$
5: $\quad m'(\mathcal{Q}_n) := \kappa$
6: $\quad$ Πρόσθεσε την $\langle \mathcal{Q}_c, \mathcal{Q}_n \rangle$ στον $\mathcal{G}'$
7: $\quad$ **for** $\sigma \in$ mergeCQs$(\mathcal{Q}_\alpha\kappa, \mathcal{Q})$ **do**
8: $\quad\quad$ Πρόσθεσε την $\langle \mathcal{Q}_n, \mathcal{Q}_\sigma \rangle$ στον $\mathcal{G}'$
9: $\quad\quad$ **for all** $\mathcal{Q}'$ s.t. $\mathcal{Q} \rightsquigarrow_{\mathcal{G}} \mathcal{Q}'$ **do**
10: $\quad\quad\quad$ **if** dom$(\sigma) \subseteq$ var$(\mathcal{Q}') \cup$ dom$(m(\mathcal{Q}'))$ **then**
11: $\quad\quad\quad\quad \mu' :=$ buildSubst$(\sigma, m(\mathcal{Q}'))$
12: $\quad\quad\quad\quad m'([\mathcal{Q}' \rightarrow \mathcal{Q}_A(nv)]\mu') := \mu'$
13: $\quad\quad\quad\quad$ **for all** $\langle \mathcal{Q}', \mathcal{Q}'' \rangle \in \mathcal{G}$ **do**
14: $\quad\quad\quad\quad\quad \mu'' :=$ buildSubst$(\sigma, m(\mathcal{Q}''))$
15: $\quad\quad\quad\quad\quad$ Πρόσθεσε την $\langle [\mathcal{Q}' \rightarrow \mathcal{Q}_A(nv)]\mu', [\mathcal{Q}'' \rightarrow \mathcal{Q}_A(nv)]\mu'' \rangle$ στον $\mathcal{G}'$
16: $\quad\quad\quad\quad$ **end for**
17: $\quad\quad\quad$ **end if**
18: $\quad\quad$ **end for**
19: $\quad$ **end for**
20: **end if**

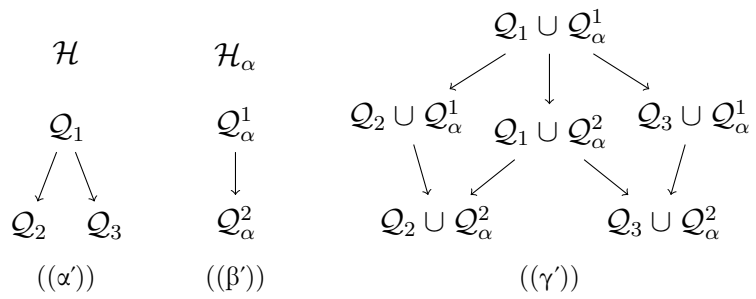
ζεται με τα ακόλουθα βήματα:

1. Ορίζει $\mu := \kappa \cup \sigma$
2. Για κάθε $z \mapsto y \in \sigma$ εάν υπάρχει η $z \mapsto y'$ στο κ τότε αντικαθιστά την $z \mapsto y'$ στο μ με την $y \mapsto y'$.

$\triangle$

Αξίζει να σημειωθεί πως η συνάρτηση αυτή είναι ιδιαίτερα σημαντική όταν για ένα ερώτημα $\mathcal{Q}'$ που πρόκειται να αντιγραφεί έχουμε $m(\mathcal{Q}') \neq \emptyset$.

Μετά από τον έλεγχο για το εάν ένα ερώτημα μπορεί να συγχωνευθεί, ο αλγόριθμος προσπελαύνει τα ερωτήματα που παράγονται από το $\mathcal{Q}_h$ (γραμμές 27–30) καθώς και αυτά που παράγονται από το $\mathcal{Q}_\alpha$ στον $\mathcal{G}_\alpha$ (γραμμές 31–34) και κατασκευάζει κατάλληλους διάδοχους για το $\mathcal{Q}_c$. Αυτό γίνεται με τη χρήση της υπο-ρουτίνας buildChildren, που απεικονίζεται στον Αλγόριθμο 12 και ακολουθεί μια παρόμοια προσέγγιση με τα προηγούμενα. Υπολογίζει, δηλαδή την ένωση μεταξύ του $\mathcal{Q}_h$ (ενός παιδιού του $\mathcal{Q}_h$) και κάποιου παιδιού του $\mathcal{Q}_\alpha$ (και του $\mathcal{Q}_\alpha$) και στη συνέχεια επιπλέον ελέγχει εάν υπάρχει κάποιο σημείο-συγχώνευσης. Εάν υπάρχει κάποιο τέτοιο σημείο τότε αντιγράφει τον σχετικό υπογράφο στο αποτέλεσμα. Τέλος, τα παιδιά του $\mathcal{Q}_h$ ($\mathcal{Q}_\alpha$) προστίθενται στην Q (Q_α).



Σχήμα 4.1: Γράφοι επανεγγραφής για το Παράδειγμα 4.2.2.

Αξίζει να σημειωθεί ότι οι $\mathcal{G}$ και $\mathcal{G}_\alpha$ μπορούν να είναι κυκλικοί. Συνεπώς ο αλγόριθμος χρειάζεται να σημειώνει ποιούς κόμβους έχει επισκεφθεί (αντιγράψει) έτσι ώστε να αποφύγει να τους επισκεφθεί (αντιγράψει) ξανά. Κάτι τέτοιο μπορεί να επιτευχθεί χρησιμοποιώντας γνωστές τεχνικές διάσχισης γράφων.

Παράδειγμα 4.2.2. Έστω οι γράφοι επανεγγραφής:

$$\mathcal{G} = \langle \{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3\}, \mathcal{H}, m\rangle \quad \text{και} \quad \mathcal{G}_\alpha = \langle \{\mathcal{Q}^1_\alpha, \mathcal{Q}^2_\alpha\}, \mathcal{H}_\alpha, m_\alpha\rangle$$

όπου οι $\mathcal{H}$ και $\mathcal{H}_\alpha$ φαίνονται στα Σχήματα 4.1(α′) και 4.1(β′), αντίστοιχα. Υποθέτουμε επίσης ότι $m(\mathcal{Q}_i) = \emptyset$ για κάθε $1 \leq i \leq 3$ και ότι όλα τα ερωτήματα $\mathcal{Q}_i$ μπορούν να επεκταθούν. Το Σχήμα 4.1(γ′) απεικονίζει τον γράφο που παράγεται από τον Αλγόριθμο 9 για τους γράφους $\mathcal{G}$ και $\mathcal{G}_\alpha$, υποθέτωντας ότι για κάθε $\mathcal{Q}^j_\alpha \in u_\alpha$ και για κάθε $\mathcal{Q}_i \in u$, έχουμε $\mathsf{mergeCQs}(\mathcal{Q}^j_\alpha, \mathcal{Q}_i) = \emptyset$—ήτοι, δεν προκύπτουν συγχωνεύσεις. Πιο συγκεκριμένα, έχουμε τα παρακάτω βήματα:

- Στην πρώτη επανάληψη έχουμε $\mathcal{Q} = \{\mathcal{Q}_1\}$ και $\mathcal{Q}_\alpha = \{\mathcal{Q}^1_\alpha\}$, συνεπώς ισχύει $\mathcal{Q}_h = \mathcal{Q}_1$ και $\mathcal{Q}_\alpha = \mathcal{Q}^1_\alpha$. Στη γραμμή 11, παράγεται το ΣΕ $\mathcal{Q}_1 \cup \mathcal{Q}^1_\alpha$ (αφού $m(\mathcal{Q}_1) = \emptyset$). Στη συνέχεια, στον βρόχο επανάληψης στις γραμμές 27–30, κατασκευάζονται τα ερωτήματα $\mathcal{Q}_2 \cup \mathcal{Q}^1_\alpha$ και $\mathcal{Q}_3 \cup \mathcal{Q}^1_\alpha$, ορίζονται ως παιδιά του $\mathcal{Q}_1 \cup \mathcal{Q}^1_\alpha$ και τα ερωτήματα $\mathcal{Q}_2, \mathcal{Q}_3$ προστίθενται στην Q. Στη συνέχεια, στον επαναληπτικό βρόχο στις γραμμές 31–34 κατασκευάζεται το ερώτημα $\mathcal{Q}_1 \cup \mathcal{Q}^2_\alpha$, ορίζεται ως παιδί της $\mathcal{Q}_1 \cup \mathcal{Q}^1_\alpha$ και το ερώτημα $\mathcal{Q}^2_\alpha$ προστίθεται στην Q_α.

- Στην επόμενη επανάληψη έχουμε $Q_\alpha = \{\mathcal{Q}^2_\alpha\}$, συνεπώς επιλέγεται η $\mathcal{Q}^2_\alpha$ και τώρα $\mathcal{Q}_h = \mathcal{Q}_1$ και $\mathcal{Q}_\alpha = \mathcal{Q}^2_\alpha$. Τότε, στη γραμμή 11, κατασκευάζεται το ΣΕ $\mathcal{Q}_1 \cup \mathcal{Q}^2_\alpha$ και στη συνέχεια, με παρόμοιο τρόπο όπως περιγράφηκε παραπάνω, κατασκευάζονται τα ΣΕ $\mathcal{Q}_2 \cup \mathcal{Q}^2_\alpha$ και $\mathcal{Q}_3 \cup \mathcal{Q}^2_\alpha$ και τα ζεύγη $\langle \mathcal{Q}_1 \cup \mathcal{Q}^2_\alpha, \mathcal{Q}_2 \cup \mathcal{Q}^2_\alpha\rangle$ και $\langle \mathcal{Q}_1 \cup \mathcal{Q}^2_\alpha, \mathcal{Q}_3 \cup \mathcal{Q}^2_\alpha\rangle$ προστίθενται στο αποτέλεσμα.

- Στην επόμενη επανάληψη, ο αλγόριθμος επιλέγει το $\mathcal{Q}_2$ από την Q και η Q_α αρχικοποιείται ξανά στο $\{\mathcal{Q}^1_\alpha\}$. Ο αλγόριθμος, τότε θα κατασκευάσει πάλι το ερώτημα $\mathcal{Q}_2 \cup \mathcal{Q}^1_\alpha$ και θα το «συνδέσει» με το ερώτημα $\mathcal{Q}_2 \cup \mathcal{Q}^2_\alpha$ που έχει κατασκευαστεί προηγούμενα.

- Τέλος, ο αλγόριθμος επιλέγει το $\mathcal{Q}_3$ από την Q κατασκευάζει πάλι το $\mathcal{Q}_3 \cup \mathcal{Q}^1_\alpha$ και το «συνδέει» με το $\mathcal{Q}_3 \cup \mathcal{Q}^2_\alpha$. Στη συνέχεια, ο αλγόριθμος τερματίζει. $\diamondsuit$

4.2.1 Ορθότητα

Στην ενότητα αυτή δείχνουμε την ορθότητα του Αλγορίθμου 9.

Εξ ορισμού της συνάρτησης mergeCQs φαίνεται ότι ο αλγόριθμος μας δεν χρησιμοποιεί το πρωτότυπο βήμα μείωσης. Για παράδειγμα, για το ΣΕ $R(x,y), R(y,z) \rightarrow \mathcal{Q}_A(x)$ το πρότυπο βήμα μείωσης θα παράγει το ερώτημα $R(x,x) \rightarrow \mathcal{Q}_A(x)$ με τον πιο γενικό ενοποιητή $\{z \mapsto y, y \mapsto x\}$. Παρόλα αυτά, για τα $\mathcal{Q}_1 := R(x,y) \rightarrow \mathcal{Q}_A(x)$ και $\mathcal{Q}_2 := R(y,z) \rightarrow \mathcal{Q}_A(x)$, έχουμε ότι mergeCQs$(\mathcal{Q}_2, \mathcal{Q}_1) = \emptyset$ και συνεπώς το $\mathcal{Q}_2$ δεν συγχωνεύεται με το $\mathcal{Q}_1$. Η ορθότητα του περιορισμένου βήματος μείωσης φαίνεται από το παρακάτω Λήμμα.

Λήμμα 4.2.3. *Έστω $\mathcal{Q}_1$, $\mathcal{Q}_2$ ΣΕ τ.ω. το $\mathcal{Q}_2$ να υπάγει το $\mathcal{Q}_1$ και έστω $\mathcal{Q}'_1$ το αποτέλεσμα της εφαρμογής ενός αξιώματος I στο $\mathcal{Q}_1$. Εάν το $\mathcal{Q}_2$ δεν υπάγει το $\mathcal{Q}'_1$, τότε είτε το I είναι εφαρμόσιμο στο $\mathcal{Q}_2$ και το αποτέλεσμα υπάγει το $\mathcal{Q}'_1$ ή για $i \geq 1$, υπάρχουν στο $\mathcal{Q}_2$ άτομα της μορφής $P(u,v), P(z_i,v)$ ή $P(v,u), P(v,z_i)$ τ.ω. για $\lambda := \{z_i \mapsto u \mid i \geq 1\}$, το I είναι εφαρμόσιμο στο $\mathcal{Q}_2\lambda$ και το αποτέλεσμα υπάγει το $\mathcal{Q}'_1$.*

Απόδειξη: Εφόσον το $\mathcal{Q}_2$ υπάγει το $\mathcal{Q}_1$, υπάρχει μια αντικατάσταση θ τ.ω. $\mathcal{Q}_2\theta \subseteq \mathcal{Q}_1$. Συνεπώς, αφού το $\mathcal{Q}'_1$ είναι το αποτέλεσμα της εφαρμογής του I στο $\mathcal{Q}_1$ τα προηγούμενα συνεπάγονται ότι το αξίωμα I εφαρμόζεται σε ένα άτομο $\alpha \in \mathcal{Q}_1$ που υπάρχει επίσης στο $\mathcal{Q}_2\theta$ αντικαθιστώντας το με ένα άτομο β που δεν εμφανίζεται στο $\mathcal{Q}_2\theta$. Συνεχίζοντας, το γεγονός ότι $\alpha \in \mathcal{Q}_2\theta$ συνεπάγεται ότι το $\mathcal{Q}_2$ περιέχει ένα άτομο α_0 τέτοιο ώστε $\alpha = \alpha_0\theta$, ήτοι, ένα άτομο με το ίδιο όνομα κατηγορήματος με το α, που πιθανότατα περιέχει διαφορετικές μεταβλητές. Με χρήση μελέτης περιπτώσεων στους διαφορετικούς τύπους εφαρμογής ανάμεσα σε αξιώματα και ατόμων ερωτημάτων δείχνουμε ότι είτε το I εφαρμόζεται επίσης στο άτομο α_0 και για $\mathcal{Q}'_2$ το αποτέλεσμα της εφαρμογής του I στο $\mathcal{Q}_2$, η θ μπορεί να επεκταθεί σε μια νέα αντικατάσταση σ τ.ω. $\mathcal{Q}'_2\sigma \subseteq \mathcal{Q}'_1$, ή ότι για $i \geq 1$ υπάρχουν στο $\mathcal{Q}_2$ άτομα της μορφής $P(u,v), P(z_i,v)$ ή $P(v,u), P(v,z_i)$ τέτοια ώστε για $\lambda := \{z_i \mapsto u \mid i \geq 1\}$, το I είναι εφαρμόσιμο στο $\mathcal{Q}_2\lambda$ και το αποτέλεσμα υπάγει το $\mathcal{Q}'_1$.

Ας θυμηθούμε πρώτα ότι $\alpha \in \mathcal{Q}_1$, $\alpha \in \mathcal{Q}_2\theta$, $\alpha_0 \in \mathcal{Q}_2$, και $\beta \in \mathcal{Q}'_1$. Διακρίνουμε τις παρακάτω περιπτώσεις:

1. Εάν $\alpha = A(x)$, τότε το αξίωμα I είναι της μορφής $C(x) \rightarrow A(x)$, με C μια DL-Lite$_R$-έννοια, το β είναι της μορφής $C(x)$, ενώ το α_0 είναι της μορφής $A(u)$, με $u \mapsto x \in \theta$. Τότε, το I είναι επίσης εφαρμόσιμο και στο $\mathcal{Q}_2$ κατασκευάζοντας ένα νέο ερώτημα $\mathcal{Q}'_2$ που περιέχει το $C(u)$. Τότε, διακρίνουμε τις παρακάτω περιπτώσεις:

   - Εάν το αξίωμα I είναι της μορφής $B(x) \rightarrow A(x)$, τότε $\beta = B(x)$. Στην περίπτωση αυτή, $B(u) \in \mathcal{Q}'_2$, και για $\sigma := \theta$ έχουμε $B(x) \in \mathcal{Q}'_2\sigma$ το οποίο συνεπάγεται ότι $\mathcal{Q}'_2\sigma \subseteq \mathcal{Q}'_1$.
   - Εάν το αξίωμα I είναι της μορφής $\exists y.P(x,y) \rightarrow A(x)$, τότε $\beta = P(x,y)$ για μια νέα μεταβλητή y στο $\mathcal{Q}_1$. Στην περίπτωση αυτή, ισχύει $P(u,z) \in \mathcal{Q}'_2$ για μια νέα μεταβλητή z στο $\mathcal{Q}_2$. Εφόσον η z είναι νέα στο $\mathcal{Q}_2$, η θ μπορεί να επεκταθεί στην $\sigma := \theta \cup \{z \mapsto y\}$. Τότε, ισχύει $P(x,y) \in \mathcal{Q}'_2\sigma$ που συνεπάγεται ότι $\mathcal{Q}'_2\sigma \subseteq \mathcal{Q}'_1$.

- Η περίπτωση που το αξίωμα I είναι της μοφής $\exists y.P(y,x) \rightarrow A(x)$ είναι συμμετρική με την προηγούμενη.

2. Εάν $\alpha = P(x,y)$ και το I είναι της μορφής $S(x,y) \rightarrow P(x,y)$, τότε $\beta = S(x,y)$, ενώ το α_0 είναι της μορφής $P(u,v)$ με $\{u \mapsto x, v \mapsto y\} \subseteq \theta$. Τότε, το I είναι επίσης εφαρμόσιμο στο $\mathcal{Q}_2$ και η εφαρμογή του έχει ως αποτέλεσμα τη δημιουργία του $\mathcal{Q}'_2$ με $S(u,v) \in \mathcal{Q}'_2$, ενώ για $\sigma := \theta$ έχουμε $S(x,y) \in \mathcal{Q}'_2\sigma$ που συνεπάγεται ότι $\mathcal{Q}'_2\sigma \subseteq \mathcal{Q}'_1$. Η περίπτωση όπου το I είναι της μορφής $S(y,x) \rightarrow P(x,y)$, είναι συμμετρική.

3. Εάν $\alpha = P(x,y)$, η y είναι μη-δεσμευμένη στο $\mathcal{Q}_1$, και το I είναι της μορφής $C(x) \rightarrow \exists y.P(x,y)$ με την C να είναι μια DL-Lite$_R$-έννοια, τότε $\beta = C(x)$ και $\alpha_0 = P(u,v)$ με $\{u \mapsto x, v \mapsto y\} \subseteq \theta$. Εάν η v είναι επίσης μη-δεσμευμένη στο $\mathcal{Q}_2$, τότε το I είναι εφαρμόσιμο στο $\mathcal{Q}_2$ και για το $\mathcal{Q}'_2$ το αποτέλεσμα θα περιέχει το $A(u) \in \mathcal{Q}'_2$. Συνεπώς, για $\sigma := \theta$ θα έχουμε $A(x) \in \mathcal{Q}'_2\sigma$ και επίσης $\mathcal{Q}'_2\sigma \subseteq \mathcal{Q}'_1$. Εάν η v είναι δεσμευμένη στο $\mathcal{Q}_2$ αυτό συνεπάγεται ότι υπάρχει κάποιο άλλο άτομο α_2 στο $\mathcal{Q}_2$ στο οποίο αναφέρεται η v. Το α_2 δεν μπορεί να είναι μια έννοια της μορφής $A(v)$ επειδή εξαιτίας του $\mathcal{Q}_2\theta \subseteq \mathcal{Q}_1$ θα είχαμε ότι $A(v)\theta = A(y) \in \mathcal{Q}_1$ και συνεπώς η y θα ήταν δεσμευμένη στο $\mathcal{Q}_1$ γεγονός που θα οδηγούσε σε αντίφαση (θα εμφανιζόταν και στα $A(y)$ και $P(x,y)$). Άρα, το α_2 πρέπει να είναι άτομο ρόλου. Όμως, το α_2 δεν μπορεί να είναι της μοφής $P(v,z)$, για μια οποιαδήποτε μεταβλητή z καθώς τότε, πάλι η $\mathcal{Q}_2\theta \subseteq \mathcal{Q}_1$ συνεπάγεται ότι $P(v,z)\theta = P(y,z\theta) \in \mathcal{Q}_1$ και συνεπώς η y θα ήταν δεσμευμένη στο $\mathcal{Q}_1$. Η μόνη πιθανή περίπτωση είναι το α_2 να είναι της μορφής $P(z,v)$ με $z \mapsto x \in \theta$ (πάλι εάν η z αντιστοιχηστεί σε μια μεταβλητή διάφορη της x, αυτό θα συνεπαγόταν ότι το $\mathcal{Q}_1$ έχει δύο ρόλους που έχουν την y σαν δεύτερο όρισμα και η y θα ήταν δεσμευμένη στο $\mathcal{Q}_1$). Συνεπώς, έχουμε $\alpha_0\theta = P(u,v)\theta = P(z,v)\theta = a_2\theta$, που συνεπάγεται ότι τα δύο άτομα ενοποιούνται. Έστω τώρα η αντικατάσταση $\sigma_1 := \{z \mapsto u\}$. Όπως έχει αναφερθεί προηγουμένως, η θ αντίστοιχεί τόσο την z όσο και την u στην x, συνεπώς, για την ορισμένη αντικατάσταση σ_1 έχουμε $\mathcal{Q}_2\sigma_1 \circ \theta = \mathcal{Q}_2\theta$ και άρα το $\mathcal{Q}_2\sigma_1$ υπάγει επίσης το $\mathcal{Q}_1$. Συνεπώς, όλες οι προηγούμενες συνθήκες εφαρμόζονται στο $\mathcal{Q}_2\sigma_1$—δηλαδή, είτε το $\mathcal{Q}_2\sigma_1$ υπάγει το $\mathcal{Q}'_1$, είτε το I είναι εφαρμόσιμο στο $\mathcal{Q}_2\sigma_1$ και το αποτέλεσμα υπάγει το $\mathcal{Q}'_1$. Εάν δεν ισχύει τίποτα από αυτά τότε πάλι το άτομο $P(z',v)$ με $z' \neq z$ πρέπει να υπάρχει στο $\mathcal{Q}_2\sigma_1$, έτσι ώστε να ενοποιείται με το $P(u,v)$ και η σ_2 μπορεί να κατασκευαστεί έτσι ώστε το $\mathcal{Q}_2\sigma_1 \circ \sigma_2$ να υπάγει το $\mathcal{Q}_1$. Εφόσον υπάρχει πεπερασμένος αριθμός ατόμων σε ένα ερώτημα, σε κάποιο σημείο κάποια αντικατάσταση $\lambda := \{z_i \mapsto u \mid i \geq 1\}$ θα υπάρχει τέτοια ώστε το I να είναι εφαρμόσιμο στο $\mathcal{Q}_2\lambda$ και το αποτέλεσμα θα υπάγει το $\mathcal{Q}'_1$. Αξίζει επίσης να σημειωθεί ότι κάθε αντικατάσταση σ_i είναι ανεξάρτητη και συνεπώς μπορούν να εφαρμοστούν με οποιαδήποτε σειρά στο $\mathcal{Q}_2$.

4. Εάν $\alpha = P(y,x)$, η y είναι μη-δεσμευμένη στο $\mathcal{Q}_1$, και το I είναι της μορφής $C(x) \rightarrow \exists y P(y,x)$, τότε $\beta = C(x)$ και $\alpha_0 = P(v,u)$ με $\{v \mapsto y, u \mapsto x\} \subseteq \theta$. Ακολουθώντας μια αντίστοιχη συλλογιστική με την προηγούμενη περίπτωση

μπορούμε να συμπεράνουμε ότι, είτε η v είναι επίσης μη-δεσμευμένη στο $\mathcal{Q}_2$ και άρα το I είναι εφαρμόσιμο στο $\mathcal{Q}_2$ και το αποτέλεσμα υπάγει το $\mathcal{Q}'_1$, είτε δεν είναι και σε αυτή την περίπτωση μια λίστα ατόμων $P(v, z_i)$ υπάρχει έτσι ώστε για $\lambda := \{z_i \mapsto u\}$, το I να είναι εφαρμόσιμο στο $\mathcal{Q}_{2\lambda}$ και το αποτέλεσμα να υπάγει το $\mathcal{Q}'_1$. □

Όπως έχει εξηγηθεί ήδη στην Ενότητα 4.1, το βήμα μείωσης είχε εισαχθεί επειδή ένα αξίωμα I μπορεί να είναι εφαρμόσιμο μόνο σε μια μείωση $\mathcal{Q}'$ ενός ΣΕ $\mathcal{Q}$. Παρόλα αυτά, είναι ευρέως γνωστό ότι εάν το $\mathcal{Q}'$ αποτελεί μείωση του $\mathcal{Q}$, τότε το $\mathcal{Q}$ υπάγει το $\mathcal{Q}'$. Συνεπώς, εάν κάποιο αξίωμα I είναι εφαρμόσιμο στο $\mathcal{Q}'$ αλλά όχι στην $\mathcal{Q}$, το Λήμμα 4.2.3 υπαγορεύει ότι μόνο ένας περιορισμένος τύπος μείωσης πάνω σε άτομα στο $\mathcal{Q}$ (το περιορισμένο βήμα μείωσης) είναι ικανός να εξασφαλιστεί ένα ΣΕ στο οποίο να είναι εφαρμόσιμο το I. Οι μειώσεις αυτές είναι εκείνες ακριβώς οι μειώσεις που χρησιμοποιούνται για να αποφασίσουμε εάν ένα ερώτημα από το $\mathcal{R}_\alpha$ μπορεί να συγχωνευθεί με ένα ερώτημα στην επανεγγραφή αναφοράς.

Έστω ένα ΣΕ $\mathcal{Q}$, ένα TBox $\mathcal{T}$, ένα άτομο α, ένας γράφος επανεγγραφής $\mathcal{G}$ για τα $\mathcal{Q}, \mathcal{T}$ και ένας γράφος $\mathcal{G}_\alpha$ που υπολογίζεται με χρήση του Αλγορίθμου 9 στην γραμμή 1. Έστω επίσης ότι $\mathcal{R}_Q$ είναι η επανεγγραφή ΣΣΕ που υπολογίζεται με χρήση του προτότυπου αλγορίθμου PerfectRef για τα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q}), \mathcal{T}$. Για να δείξουμε ότι το περιορισμένο μήμα μείωσης είναι σωστό, γεγονός που συνεπάγεται και ότι ο Αλγόριθμος 8 είναι σωστός θα χρησιμοποιήσουμε επαγωγή στον αριθμό των βημάτων για τα οποία έχει εκτελεστεί ο αλγόριθμος PerfectRef για τα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q}), \mathcal{T}$. Για παράδειγμα, έστω ότι $\mathcal{R}_i \subseteq \mathcal{R}_Q$ είναι η επανεγγραφή ΣΣΕ που υπολογίζεται στο βήμα i and that for every $\mathcal{Q}_i \in \mathcal{R}_i$ υπάρχουν κάποιο $\mathcal{Q}_h$ στο $\mathcal{G}$ και κάποιο $\mathcal{Q}_\alpha$ στον $\mathcal{G}_\alpha$ τέτοια ώστε $\bigwedge \mathsf{bd}(\mathcal{Q}_h) \cup \mathsf{bd}(\mathcal{Q}_\alpha\kappa) \rightarrow \mathcal{Q}_A(nv)$, όπου $\kappa = m(\mathcal{Q}_h)$ και το $nv = \mathsf{avar}(\mathcal{Q}_h) \cup (\mathsf{var}(\mathcal{Q}_\alpha\kappa) \cap \mathsf{avar}(\mathcal{Q}))$ υπάγει το $\mathcal{Q}_i$. Έστω ότι στο βήμα $i+1$ κάποιο αξίωμα I εφαρμόζεται στο $\mathcal{Q}_i$ παράγοντας το $\mathcal{Q}_{i+1}$. Από το Λήμμα 4.2.3 έχουμε ότι είτε το I είναι εφαρμόσιμο στο $\bigwedge \mathsf{bd}(\mathcal{Q}_h) \cup \mathsf{bd}(\mathcal{Q}_\alpha\kappa) \rightarrow \mathcal{Q}_A(nv)$ και το αποτέλεσμα υπάγει το $\mathcal{Q}_{i+1}$ ή ότι μια σειρά από περιορισμένα βήματα μείωσης είναι εφαρμόσιμα. Στην πρώτη περίπτωση το I είτε είναι εφαρμόσιμο σε κάποιο άτομο στο $\mathcal{Q}_h$ παράγοντας κάποιο ερώτημα $\mathcal{Q}'_h$ για το οποίο θα έχουμε $\langle \mathcal{Q}_h, \mathcal{Q}'_h \rangle \in \mathcal{G}$ ή το I είναι εφαρμόσιμο στο $\mathcal{Q}_\alpha\kappa$. Συνεπώς, στην πρώτη περίπτωση έχουμε ότι κάποιο $\mathcal{Q}'_h$ στο $\mathcal{G}$ τέτοιο ώστε $\bigwedge \mathsf{bd}(\mathcal{Q}'_h) \cup \mathsf{bd}(\mathcal{Q}_\alpha\kappa) \rightarrow \mathcal{Q}_A(nv)$ υπάγει το $\mathcal{Q}_{i+1}$. Στην δεύτερη περίπτωση η ύπαρξη κάποιου ερωτήματος $\mathcal{Q}'_\alpha$ στον $\mathcal{G}_\alpha$ έτσι ώστε $\langle \mathcal{Q}_\alpha, \mathcal{Q}'_\alpha \rangle \in \mathcal{G}_\alpha$ και το $\bigwedge \mathsf{bd}(\mathcal{Q}_h) \cup \mathsf{bd}(\mathcal{Q}'_\alpha\kappa) \rightarrow \mathcal{Q}_A(nv)$ να υπάγει το $\mathcal{Q}_{i+1}$ προκύπτει από το παρακάτω λήμμα.

Λήμμα 4.2.4. *Έστω $\mathcal{Q}$ ΣΕ που περιέχει ακριβώς ένα άτομο στο σώμα του και έστω κ αντικατάσταση έτσι ώστε κάποιο αξίωμα I να είναι εφαρμόσιμο στο $\mathcal{Q}\kappa$ παράγοντας το $\mathcal{Q}'$. Τότε το I είναι επίσης εφαρμόσιμο και στο $\mathcal{Q}$ και για το αποτέλεσμα $\mathcal{Q}''$ έχουμε $\mathcal{Q}''\kappa = \mathcal{Q}'$ (μόντουλο τη μετονομασία των νέων μεταβλητών).*

Απόδειξη: Εάν η κ δεν αλλάζει κάποια από τις μεταβλητές του $\mathcal{Q}$ τότε η ιδιότητα αυτή προκύπτει τετριμένα. Εάν αλλάζει, τότε δείχνουμε την ιδιότητα αυτή παρουσιάζοντας μια ανάλυση περιπτώσεων για τη μορφή του $\mathcal{Q}$.

1. Το $\mathcal{Q}$ είναι της μορφής $A(x) \rightarrow \mathcal{Q}_A(x)$. Τότε, η κ μπορεί να είναι της μορφής $\{x \mapsto y\}$. Διακρίνουμε τις παρακάτω περιπτώσεις ανάλογα με την μορφή του I:

   (α′) Εάν το I είναι της μορφής $C(x) \rightarrow A(x)$, τότε $\mathcal{Q}' = C(y) \rightarrow \mathcal{Q}_A(y)$, ενώ το I είναι επίσης εφαρμόσιμο στο $\mathcal{Q}$ και το αποτέλεσμα είναι $\mathcal{Q}'' = C(x) \rightarrow \mathcal{Q}_A(x)$, οπότε προφανώς $\mathcal{Q}''\kappa = \mathcal{Q}'$.

   (β′) Εάν το I είναι της μορφής $\exists y.R(x,y) \rightarrow A(x)$, τότε $\mathcal{Q}' = R(y,z_1) \rightarrow \mathcal{Q}_A(y)$ και $\mathcal{Q}'' = R(x,z_2) \rightarrow \mathcal{Q}_A(x)$, για z_1, z_2 νέες μεταβλητές, και πάλι $\mathcal{Q}''\kappa = \mathcal{Q}'$.

2. Το $\mathcal{Q}$ είναι της μορφής $R(x,y) \rightarrow \mathcal{Q}_A(x)$. Τότε έχουμε τις παρακάτω περιπτώσεις ανάλογα με τη μορφή του I:

   (α′) Εάν το I είναι της μορφής $C(x) \rightarrow \exists y.R(x,y)$, τότε η κ δεν μπορεί να αντιστοιχήσει την y στην x (και αντίστροφα), επειδή κάτι τέτοιο θα έκανε την x (y) δεσμευμένη στο $\mathcal{Q}\kappa$. Συνεπώς η κ (πιθανότατα) αντιστοιχεί την x και/ή την y σε διαφορετικές μεταβλητές. Σε κάθε περίπτωση $\mathcal{Q}' = C(x\kappa) \rightarrow \mathcal{Q}_A(x\kappa)$. Προφανώς, το I είναι εφαρμόσιμο στο $\mathcal{Q}$ και πάλι όπως στην περίπτωση 1. μπορούμε να εξάγουμε ότι $\mathcal{Q}''\kappa = \mathcal{Q}'$.

   (β′) Εάν το I είναι της μορφής $P(x,y) \rightarrow R(x,y)$ και η κ είναι όπως στην προηγούμενη περίπτωση, τότε ο ισχυρισμός προκύπτει με παρόμοια επιχειρηματολογία. Παρόλα αυτά, η κ μπορεί να είναι της μορφής $\{y \mapsto x\}$ (ή $\{x \mapsto y\}$). Τότε, $\mathcal{Q}' = P(x,x) \rightarrow \mathcal{Q}_A(x)$ (ή $\mathcal{Q}' = P(y,y) \rightarrow \mathcal{Q}_A(y)$). Προφανώς, το I είναι επίσης εφαρμόσιμο στο $\mathcal{Q}$ και το αποτέλεσμα είναι το $\mathcal{Q}'' = P(x,y) \rightarrow \mathcal{Q}_A(x)$ για το οποίο $\mathcal{Q}''\kappa = P(x,x) \rightarrow \mathcal{Q}_A(x)$ (ή $\mathcal{Q}''\kappa = P(y,y) \rightarrow \mathcal{Q}_A(y)$). Σε κάθε περίπτωση $\mathcal{Q}''\kappa = \mathcal{Q}'$.

3. Το $\mathcal{Q}$ είναι της μορφής $R(x,y) \rightarrow \mathcal{Q}_A(x,y)$ και το I είναι της μορφής $P(x,y) \rightarrow R(x,y)$. Η περίπτωση αυτή είναι παρόμοια με την 2β′.

Τέλος, το θεώρημα που ακολουθεί δείχνει την ορθότητα και την πληρότητα του αλγορίθμου. Η απόδειξη που δίνεται στη συνέχεια αποτελεί μια συστηματική μελέτη όλων των περιπτώσεων που έχουν αναφερθεί προηγούμενα.

Θεώρημα 4.2.5. *Έστω $\mathcal{Q}$ ΣΕ, έστω $\mathcal{T}$ DL-Lite$_R$-TBox, έστω α άτομο έτσι ώστε* $\mathsf{var}(\alpha) \cap \mathsf{var}(\mathcal{Q}) \neq \emptyset$ *και έστω $\mathcal{G}$ γράφος επανεγγραφής για τα $\mathcal{Q}, \mathcal{T}$ που είναι κλειστός ως προς την αναδιατύπωση για τα $\mathcal{Q}, \mathcal{T}$. Έστω $\mathcal{G}'$ ο γράφος που επιστρέφεται από τον Αλγόριθμο 9 όταν σε αυτόν δίνονται ως είσοδος τα $\mathcal{G}, \alpha$ και $\mathcal{T}$. Τότε ο $\mathcal{G}'$ είναι ένας γράφος επανεγγραφής για τα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q}), \mathcal{T}$ που είναι κλειστός ως προς την αναδιατύπωση για τα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q}), \mathcal{T}$.*

Απόδειξη: Πρώτα απ΄ όλα πρέπει να τονίσουμε ότι ο Αλγόριθμος ex-PerfectRef επεκτείνει τον Αλγόριθμο PerfectRef, συνεπώς για ένα δεδομένο ερώτημα $\mathcal{Q}$ και ένα DL-Lite$_R$-TBox $\mathcal{T}$ υπολογίζει ένα γράφο επανεγγραφής για τα $\mathcal{Q}, \mathcal{T}$. Επιπρόσθετα, εφόσον ο PerfectRef εφαρμόζει εξαντλητικά τα βήματα της αναδιατύπωσης και μείωσης και δεν κλαδεύει τα υπολογισμένα ερωτήματα, το αποτέλεσμα του επεκτεταμένου αλγορίθμου είναι κλειστό ως προς την αναδιατύπωση.

Έστω ότι ο γράφος επανεγγραφής $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$ είσοδος του Αλγορίθμου 9 και έστω ότι ο γράφος επανεγγραφής $\mathcal{G}' = \langle \mathcal{R}'_Q, \mathcal{H}', m' \rangle$ είναι η έξοδος του. Επιπλέον, για τη συνέχεια της απόδειξης, έστω ότι $jv = \mathsf{var}(\alpha) \cap \mathsf{avar}(\mathcal{Q})$, έστω ότι ο γράφος $\mathcal{G}_\alpha = \langle \mathcal{R}_\alpha, \mathcal{H}_\alpha, m_\alpha \rangle$ είναι ο γράφος επανεγγραφής που υπολογίζεται από τον Αλγόριθμο 9 στη γραμμή 1, και έστω ότι ο $\mathcal{G}_i$ είναι ένας γράφος επανεγγραφής που υπολογίζεται ύστερα από i βήματα του Αλγορίθμου ex-PerfectRef όταν αυτός εφαρμόζεται για τα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q}), \mathcal{T}$—δηλαδή, μετά την παραγωγή i ερωτημάτων από την εφαρμογή των βημάτων αναδιατύπωσης και μείωσης.

Η απόδειξη του θεωήματος προκύπτει από την παρακάτω ιδιότητα:

> $(\star)$: Για κάθε $i \geq 0$ και για κάθε $\mathcal{Q}_j$ στο $\mathcal{R}_i$ υπάρχει μια κορυφή $\mathcal{Q}_h$ στον $\mathcal{G}$ και μια κορυφή $\mathcal{Q}_\alpha$ στον $\mathcal{G}_\alpha$ τέτοιες ώστε, για $\kappa = m(\mathcal{Q}_h)$ και $nv = \mathsf{avar}(\mathcal{Q}_h) \cup (\mathsf{avar}(\mathcal{Q}) \cap \mathsf{var}(\mathcal{Q}_\alpha\kappa))$ έχουμε $\mathsf{canBeJoined}(\mathcal{Q}_h, \kappa, jv) =$ αληθές, και ισχύει μια από τις παρακάτω συνθήκες:
>
> 1. Το $\bigwedge \mathsf{bd}(\mathcal{Q}_h) \cup \mathsf{bd}(\mathcal{Q}_\alpha\kappa) \rightarrow \mathcal{Q}_A(nv)$ υπάγει το $\mathcal{Q}_j$, ή
> 2. για κάποια αντικατάσταση $\sigma \in \mathsf{mergeCQs}(\mathcal{Q}_\alpha\kappa, \mathcal{Q}_h)$ υπάρχει κάποιο ΣΕ $\mathcal{Q}'$ για το οποίο $\mathcal{Q}_h \rightsquigarrow_{\mathcal{G}} \mathcal{Q}'$ και το $[\bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv)]\mu'$, όπου $\mu' = \mathsf{buildSubst}(\sigma, m(\mathcal{Q}'))$, υπάγει το $\mathcal{Q}_j$.

Έστω τώρα ότι η Ιδιότητα $(\star)$ ισχύει. Θα δείξουμε ότι ο $\mathcal{G}' = \langle \mathcal{R}'_Q, \mathcal{H}', m' \rangle$ είναι ένας γράφος επανεγγραφής κλειστός ως προς την αναδιατύπωση για τα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q}), \mathcal{T}$.

Πρώτα, δειχνουμε ότι η $\mathcal{R}'_Q$ είναι μια επανεγγραφή ΣΣΕ για τα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q}), \mathcal{T}$. Έστω ότι όταν ο αλγόριθμος PerfectRef εφαρμόζεται στα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q})$ και $\mathcal{T}$ τερματίζει ύστερα από n βήματα και έστω ότι η επανεγγραφή που υπολογίζεται είναι η $\mathcal{R}_n$. Τότε όμως, από την Ιδιότητα $(\star)$, γνωρίζουμε ότι με τον τερματισμό του Αλγορίθμου 9, για κάθε ΣΕ $\mathcal{Q}_n \in \mathcal{R}_n$ είτε ένα ΣΕ της μορφής $\bigwedge \mathsf{bd}(\mathcal{Q}_h) \cup \mathsf{bd}(\mathcal{Q}_\alpha\kappa) \rightarrow \mathcal{Q}_A(nv)$ ή ένα ΣΕ της μορφής $[\bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv)]\mu'$ που υπάγει το $\mathcal{Q}_n$ έχει προστεθεί στο $\mathcal{R}'_Q$. Επιπλέον, οι συναρτήσεις canBeJoined, buildSubst, και η συνθήκη $\mathsf{dom}(\sigma) \subseteq \mathsf{var}(\mathcal{Q}') \cup \mathsf{dom}(m(\mathcal{Q}'))$ εξασφαλίζουν ότι κάθε ΣΕ που παράγεται από τον αλγόριθμο μας μπορεί να παραχθεί επίσης και από τον αλγόριθμο PerfectRef. Συνεπώς, το σύνολο $\mathcal{R}'_Q$ είναι ορθό και πλήρες, και άρα είναι επανεγγραφή ΣΣΕ για τα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q}), \mathcal{T}$.

Τώρα δείχνουμε ότι ο $\mathcal{G}'$ είναι κλειστός ως προς την αναδιατύπωση. Αρχικά, από τον ορισμό των αντιστοιχίσεων $m'(\mathcal{Q}'_n)$ για κάθε ΣΕ που προστίθεται στον $\mathcal{G}'$ προκύπτει ότι η m' ικανοποιεί τις συνθήκες του Ορισμού 4.1.5. Επιπλέον, κατά τον τερματισμό του αλγορίθμου και η σχέση $\mathcal{H}'$ ικανοποιεί την ιδιότητα του Ορισμού 4.1.5. Ακόμα, είναι εύκολο να δειχθεί ότι ο αλγόριθμος θα παράγει και θα ορίσει ως κόμβο κορυφής του $\mathcal{G}'$ το ΣΕ $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q})$ με $m'(\mathcal{Q} \cup \{\alpha\}) = \emptyset$, ενώ για όλους τους κόμβους κορυφής $\mathcal{Q}_i$ στον $\mathcal{G}'$ θα έχουμε επίσης $m(\mathcal{Q}_i) = \emptyset$ καθώς κληρονομούν την αντιστοίχιση του $\mathcal{Q}$ (θα αναφερθούμε περισσότερο σε αυτό στην απόδειξη της Ιδιότητας $(\star)$ στη συνέχεια). Επιπρόσθετα, η $\mathcal{H}'$ είναι κλειστή ως προς το περιορισμένο βήμα μείωσης. Εάν για κάποια $\mathcal{Q}_h \in \mathcal{R}_Q, \kappa = m(\mathcal{Q}_h)$ και $\mathcal{Q}_\alpha \in \mathcal{R}_\alpha$ έχουμε κάποια $\sigma \in \mathsf{mergeCQs}(\mathcal{Q}_\alpha\kappa, \mathcal{Q}_h)$, τότε ο αλγόριθμος προσθέτει πρώτα το

$\mathcal{Q}^+ = \bigwedge \mathsf{bd}(\mathcal{Q}_h) \cup \mathsf{bd}(\mathcal{Q}_\alpha\kappa) \rightarrow \mathcal{Q}_A(nv)$ στον $\mathcal{G}'$ (γραμμή 13) και στη συνέχεια προσθέτει την πλειάδα $\langle \mathcal{Q}^+, [\bigwedge \mathsf{bd}(\mathcal{Q}_h) \rightarrow \mathcal{Q}_A(nv)]\sigma\rangle$ στον $\mathcal{G}'$ (γραμμή 15). Στη συνέχεια, αντιγράφει τον υπο-γράφο της $\mathcal{H}$ κάτω από το $\mathcal{Q}_h$, αλλά η $\mathcal{H}$ είναι επίσης κλειστή ως προς την αναδιατύπωση. Η μόνη ενδιαφέρουσα περίπτωση είναι για κάποιο ερώτημα $\mathcal{Q}'$ για το οποίο $\mathcal{Q}_h \rightsquigarrow_{\mathcal{G}} \mathcal{Q}'$ και έχουμε $z \mapsto x \in \sigma$ και $z \mapsto x' \in m(\mathcal{Q}')$. Η τελευταία σχέση συνεπάγεται ότι υπάρχει $\mathcal{Q}''$ τέτοιο ώστε $\langle \mathcal{Q}'', \mathcal{Q}'\rangle \in \mathcal{H}$ και $R(x', y), R(z, y)$ είναι άτομα στο $\mathcal{Q}''$ ενώ $\mathcal{Q}' = \mathcal{Q}''\{z \mapsto x'\}$, ήτοι, κάθε εμφάνιση της μεταβλητής z στο $\mathcal{Q}''$ έχει ανιστοιχηστεί στην x'. Στη νέα σχέση $\mathcal{H}'$, ο αλγόριθμος θα περιέχει το ερώτημα $\mathcal{Q}''\sigma$, δηλαδή, το $\mathcal{Q}''\sigma$ θα περιέχει τα άτομα $R(x', y), R(x, y)$ αντί των $R(x, y), R(z, y)$, δηλαδή, η z θα εχει αντιστοιχηστεί στην x. Στη συνέχεια, ένα παιδί του $\mathcal{Q}''\sigma$ πρέπει να κατασκευαστεί. Για να είναι κλειστή ως προς την αναδιατύπωση η $\mathcal{H}'$, το παιδί θα πρέπει να είναι ένα ερώτημα όπου τώρα η x, και όχι η z, θα αντιστοιχεί στην x'. Αυτό πραγματοποιείται με τη συνάρτηση buildSubst. Πιο συγκεκριμένα, εφόσον $z \mapsto x \in \sigma$ και $z \mapsto x' \in m(\mathcal{Q}')$, η συνάρτηση επιστρέφει μια αντιστοίχιση μ που περιέχει την $z \mapsto x$ και την $x \mapsto x'$ που εφαρμόζεται στο $\mathcal{Q}'$ παράγοντας έτσι το κατάλληλο ερώτημα. Συνεπώς, ο γράφος $\mathcal{G}'$ είναι κλειστός ως προς την αναδιατύπωση για τα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q}), \mathcal{T}$.

Τώρα, δίνουμε την απόδειξη για την Ιδιότητα $(\star)$.

Για να δείξουμε αυτή την ιδιότητα χρησιμοποιούμε επαγωγή πάνω στον αριθμό των βημάτων i τα οποία ο αλγόριθμος ex-PerfectRef έχει ολοκληρώσει όταν εφαρμόζεται για τα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q}), \mathcal{T}$. Για ένα ΣΕ $\mathcal{Q}$ θα χρησιμοποιούμε την έκφραση $m_{\mathcal{Q}}$ για να αναφερθούμε στην αντιστοίχιση $m(\mathcal{Q})$.

Βασική Περίπτωση ($i = 0$): Από τη μια πλευρά, όταν ο αλγόριθμος ex-PerfectRef εφαρμόζεται στα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q}), \mathcal{T}$, στο βήμα 0 αρχικοποιεί ένα σύνολο $\mathcal{R}_0$ με την κορυφή $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q})$. Από την άλλη πλευρά, στην γραμμή 1 ο Αλγόριθμος 9 υπολογίζει τον $\mathcal{G}_\alpha$ για το $\mathcal{Q}_\alpha = \alpha \rightarrow \mathcal{Q}_A(\mathsf{var}(\alpha) \cap \mathsf{var}(\mathcal{Q}))$ και το $\mathcal{T}$ με τη χρήση του ex-PerfectRef και συνεπώς τουλάχιστον περιέχει το $\mathcal{Q}_\alpha$ ως κορυφή. Επιπλέον, εφόσον ο $\mathcal{G}$ είναι κλειστός ως προς την αναδιατύπωση για τα $\mathcal{Q}, \mathcal{T}$, τότε το $\mathcal{Q}$ εμφανίζεται ως κόμβος κορυφής στον $\mathcal{G}$ με $m_{\mathcal{Q}} = \emptyset$. Συνεπώς, από την κατασκευή των $\mathcal{Q}_\alpha$ και jv έχουμε ότι $\mathsf{canBeJoined}(\mathcal{Q}, m_{\mathcal{Q}}, jv) =$ αληθές και επίσης το $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \mathsf{bd}(\mathcal{Q}_\alpha m_q) \rightarrow \mathcal{Q}_A(nv)$ είναι το $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q})$. Συνεπώς, η συνθήκη 1. ικανοποιείται και άρα για το βήμα $i = 0$ η Ιδιότητα $(\star)$ ικανοποιείται.

Βήμα Επαγωγής: Υποθέτουμε ότι η Ιδιότητα $(\star)$ ισχύει στο βήμα i για κάθε $\mathcal{Q}_j \in \mathcal{R}_i$. Στη συνέχεια, υποθέτουμε ότι στο βήμα $i + 1$ ο αλγόριθμος παράγει το σύνολο $\mathcal{R}_{i+1}$ εφαρμόζοντας είτε ένα βήμα αναδιατύπωσης είτε ένα βήμα μείωσης σε κάποιο ερώτημα $\mathcal{Q}_j$ στο $\mathcal{R}_i$ παράγοντας το ερώτημα $\mathcal{Q}_{i+1}$. Επιπλέον, από την υπόθεση της επαγωγής, υπάρχουν κορυφές $\mathcal{Q}_h$ στον $\mathcal{G}$ και $\mathcal{Q}_\alpha$ στον $\mathcal{G}_\alpha$ τέτοιες ώστε να ικανοποιείται η Ιδιότητα $(\star)$—δηλαδή, για $\kappa = m_{\mathcal{Q}_h}$, έχουμε $\mathsf{canBeJoined}(\mathcal{Q}_h, \kappa, jv) =$ αληθές και είτε το ερώτημα $\bigwedge \mathsf{bd}(\mathcal{Q}_h) \cup \mathsf{bd}(\mathcal{Q}_\alpha\kappa) \rightarrow \mathcal{Q}_A(nv)$ (που θα καλείται $\mathcal{Q}^+$ στην συνέχεια) υπάγει το $\mathcal{Q}_j$, ή για κάποια αντικατάσταση $\sigma \in \mathsf{mergeCQs}(\mathcal{Q}_\alpha\kappa, \mathcal{Q}_h)$ και κάποιο $\mathcal{Q}'$ τέτοιο ώστε $\mathcal{Q}_h \rightsquigarrow_{\mathcal{G}} \mathcal{Q}'$, έχουμε ότι $[\bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv)]\mu'$ (που θα καλείται $\mathcal{Q}^m$ στη συνέχεια) υπάγει το $\mathcal{Q}_j$. Διακρίνουμε τώρα δύο περιπτώσεις ανάλογα με το εάν το $\mathcal{Q}_{i+1}$ έχει παραχθεί από ένα βήμα αναδιατύπωσης ή από ένα βήμα μείωσης.

Εάν εφαρμόστηκε στο $\mathcal{Q}_j$ ένα βήμα μείωσης για να παραχθεί το $\mathcal{Q}_{i+1}$, τότε είναι

γνωστό από τις ιδιότητες της μείωσης ότι το ερώτημα $\mathcal{Q}_j$ υπάγει το $\mathcal{Q}_{i+1}$. Συνεπώς, από την μεταβατικότητα της σχέσης υπαγωγής και την υπόθεση της επαγωγής το $\mathcal{Q}_{i+1}$ υπάγεται είτε από το $\mathcal{Q}^+$ είτε από το $\mathcal{Q}^m$. Συνεπως, η Ιδιότητα ($\star$) ικανοποιείται.

Διαφορετικά, έστω ότι στο $\mathcal{Q}_j$ εφαρμόστηκε ένα βήμα αναδιατύπωσης για την παραγωγή του $\mathcal{Q}_{i+1}$. Αυτό συνεπάγεται ότι κάποιο αξιωμα I εφαρμόζεται σε κάποιο άτομο του σώματος του $\mathcal{Q}_j$ παράγοντας το $\mathcal{Q}_{i+1}$. Από την υπόθεση της επαγωγής, είτε το $\mathcal{Q}^+$ ή το $\mathcal{Q}^m$ υπάγει το $\mathcal{Q}_j$. Εάν οποιοδήποτε από τα δύο αυτά ερωτήματα $\mathcal{Q}_{i+1}$, τότε η Ιδιότητα ($\star$) ικανοποιείται. Εάν κανένα από τα δύο ερωτήματα δεν υπάγει το $\mathcal{Q}_{i+1}$, τότε από το Λήμμα 4.2.3, είτε (i) το I είναι εφαρμόσιμο σε αυτά και το αποτέλεσμα υπάγει το $\mathcal{Q}_{i+1}$ ή (ii) για $n \geq 1$ υπάρχουν άτομα της μορφής $P(u,v), P(z_i,v)$ ή $P(v,u), P(v,z_i)$ στο $\mathcal{Q}^+$ ή στο $\mathcal{Q}^m$ έτσι ώστε για $\lambda = \{z_i \mapsto u \mid 1 \leq i \leq n\}$, το $\mathcal{Q}^+\lambda$ ή το $\mathcal{Q}^m\lambda$ υπάγει το $\mathcal{Q}_j$, το I είναι εφαρμόσιμο στο $\mathcal{Q}^+\lambda$ ή στο $\mathcal{Q}^m\lambda$ και το αποτέλεσμα υπάγει το $\mathcal{Q}_{i+1}$. Εξετάζουμε τώρα ξεχωριστά τις περιπτώσεις που το $\mathcal{Q}^+$ ή το $\mathcal{Q}^m$ υπάγουν το $\mathcal{Q}_j$.

1. Έστω ότι το $\mathcal{Q}^+$ υπάγει το $\mathcal{Q}_j$ και επίσης ότι ισχύει η περίπτωση (i) από πάνω—δηλαδή, το I είναι εφαρμόσιμο στο $\mathcal{Q}^+ = \bigwedge \mathsf{bd}(\mathcal{Q}_h) \cup \mathsf{bd}(\mathcal{Q}_\alpha\kappa) \rightarrow \mathcal{Q}_A(nv)$. Εφόσον η αναδιατύπωση εφαρμόζει κάθε φορά ένα αξίωμα σε ένα μόνο άτομο του σώματος ενός ερωτήματος, αυτό συνεπάγεται ότι το I είτε είναι εφαρμόσιμο σε κάποιο άτομο του $\mathcal{Q}_h$ και από αυτή την εφαρμογή παράγεται ένα ΣΕ της μορφής $\bigwedge \mathsf{bd}(\mathcal{Q}'_h) \cup \mathsf{bd}(\mathcal{Q}_\alpha\kappa) \rightarrow \mathcal{Q}_A(nv)$, ή είναι εφαρμόσιμο σε κάποιο άτομο του $\mathcal{Q}_\alpha\kappa$ και το αποτέλεσμα είναι ένα ΣΕ της μορφής $\bigwedge \mathsf{bd}(\mathcal{Q}_h) \cup \mathsf{bd}(\mathcal{Q}'_\alpha) \rightarrow \mathcal{Q}_A(nv)$. (Σημειώνουμε ότι το βήμα αναδιατύπωσης δεν αλλάζει τις διακεκριμένες μεταβλητές του ερωτήματος) Στην πρώτη περίπτωση, εφόσον το I είναι εφαρμόσιμο στο $\mathcal{Q}_h$ παράγοντας το $\mathcal{Q}'_h$, το $\mathcal{Q}_h$ είναι μια κορυφή στον $\mathcal{G}$ και ο $\mathcal{G}$ είναι κλειστός ως προς την αναδιατύπωση, τότε $\langle \mathcal{Q}_h, \mathcal{Q}'_h \rangle \in \mathcal{H}$ με $m_{\mathcal{Q}'_h} = \kappa$ και $\mathsf{avar}(\mathcal{Q}_h) = \mathsf{avar}(\mathcal{Q}'_h)$. Επιπλέον, $\mathsf{canBeJoined}(\mathcal{Q}'_h, m_{\mathcal{Q}'_h}, jv) = \mathsf{αληθές}$. Αυτό προκύπτει από το γεγονός ότι το I είναι εφαρμόσιμο στο $\bigwedge \mathsf{bd}(\mathcal{Q}_h) \cup \mathsf{bd}(\mathcal{Q}_\alpha\kappa) \rightarrow \mathcal{Q}_A(nv)$, συνεπώς η εφαρμογή του μπορεί να αφαιρέσει μόνο μεταβλητές που δεν εμφανίζονται στο $\mathcal{Q}_\alpha\kappa$. Συνεπώς, η συνθήκη 1. ικανοποιείται και άρα το ίδιο και η Ιδιότητα ($\star$). Μελετάμε τώρα την περίπτωση όπου το I είναι εφαρμόσιμο στο $\mathcal{Q}_\alpha\kappa$: Αρχικά, υπενθυμίζουμε ότι το $\mathcal{Q}_\alpha$ περιέχει ακριβώς ένα άτομο στο σώμα του, κάτι το οποίο προκύπτει ευθέως από το γεγονός ότι το ερώτημα για το οποίο κατασκευάζεται ο $\mathcal{G}_\alpha$ υπολογίζεται από ένα μόνο άτομο και από το γεγονός ότι ο αλγόριθμος PerfectRef παράγει πάντα ερωτήματα που περιέχουν το πολύ τον ίδιο αριθμό ατόμων με το ερώτημα από το οποίο παράγονται. Άρα, από το Λήμμα 4.2.4, το I είναι επίσης εφαρμόσιμο και στο $\mathcal{Q}_\alpha$ και για το αποτέλεσμα $\mathcal{Q}''_\alpha$ έχουμε $\mathcal{Q}''_\alpha\kappa = \mathcal{Q}'_\alpha$. Επειδή όμως $\mathcal{Q}_\alpha \in \mathcal{R}_\alpha$ και ο $\mathcal{G}_\alpha$ είναι κλειστός ως προς την αναδιατύπωση, τότε $\mathcal{Q}''_\alpha \in \mathcal{R}_\alpha$. Συνεπώς, και πάλι η συνθήκη 2. ικανοποιείται άρα και η Ιδιότητα ($\star$).

   Έστω τώρα ότι έχουμε την περίπτωση (ii) από πάνω—ήτοι, για $n \geq 1$ υπάρχουν άτομα της μορφής $P(u,v), P(z_i,v)$ ή της μορφής $P(u,v), P(z_i,v)$ στο $\mathcal{Q}^+$ τέτοια ώστε $\lambda = \{z_i \mapsto u \mid 1 \leq i \leq n\}$, το I είναι εφαρμόσιμο στο $\mathcal{Q}^+\lambda = \bigwedge \mathsf{bd}(\mathcal{Q}_h\lambda) \cup \mathsf{bd}(\mathcal{Q}_\alpha\kappa \circ \lambda) \rightarrow \mathcal{Q}_A(nv\lambda)$ και το νέο ΣΕ που παράγεται

από την εφαρμογή του I στο $\mathcal{Q}^+\lambda$, που ονομάζεται $\mathcal{Q}_f$, υπάγει το $\mathcal{Q}_{i+1}$. Θα δείξουμε τώρα πως ο αλγόριθμος μας μπορεί να παράγει τα ερωτήματα $\mathcal{Q}^+\lambda$ και $\mathcal{Q}_f$. Πρώτα σημειώνουμε ότι η ύπαρξη στο $\mathcal{Q}^+$ των ατόμων στην παραπάνω μορφή συνεπάγεται ότι μια σειρά από περιορισμένα βήματα μείωσης είναι εφαρμόσιμα ανάμεσα στα άτομα $P(z_i, v)$ και το άτομο $P(u, v)$ στο $\mathcal{Q}'_j$. Επιπλέον, εφόσον το $\mathcal{Q}_\alpha\kappa$ περιέχει μόνο ένα άτομο, οι μειώσεις αυτές είναι εφαρμόσιμες είτε ανάμεσα σε δύο άτομα του $\mathcal{Q}_h$ ή ανάμεσα σε ένα άτομο του $\mathcal{Q}_h$ και το άτομο $(q_\alpha)_\kappa$. Στην πρώτη περίπτωση, η εφαρμογή μιας τέτοιας μείωσης θα είχε ως αποτέλεσμα την παραγωγή του ΣΕ $\bigwedge \mathsf{bd}(\mathcal{Q}_h\lambda_i) \cup \mathsf{bd}(\mathcal{Q}_\alpha\kappa \circ \lambda_i) \rightarrow \mathcal{Q}_A(nv\lambda_i)$. Εφόσον ο γράφος $\mathcal{G}$ είναι κλειστός ως προς την αναδιατύπωση και από τη μορφή αυτών των μειώσεων ισχύει επίσης $\langle \mathcal{Q}_h, \mathcal{Q}'_h \rangle \in \mathcal{G}$, με $\mathcal{Q}'_h = \mathcal{Q}_h\lambda_i$ και $m_{\mathcal{Q}'_h} = \kappa \circ \lambda_i$, άρα, $\mathsf{canBeJoined}(\mathcal{Q}'_h, m_{\mathcal{Q}'_h}, jv) =$ αληθές. Εάν καμία από αυτές τις μειώσεις δεν εφαρμόζεται ανάμεσα στα $\mathcal{Q}_\alpha\kappa$ και $\mathcal{Q}_h$, τότε για την λ που έχει οριστεί όπως προηγούμενα θα έχουμε $\mathcal{Q}_h \rightsquigarrow_{\mathcal{G}} \mathcal{Q}_h\lambda$ με $m_{\mathcal{Q}_h\lambda} = \kappa \circ \lambda$ και άρα επίσης $\mathsf{canBeJoined}(\mathcal{Q}_h\lambda, m_{\mathcal{Q}_h\lambda}, jv) =$ αληθές. Τότε, το αξίωμα I είναι εφαρμόσιμο στο $\mathcal{Q}_h\lambda$ και εφόσον ο $\mathcal{G}$ είναι κλειστός ως προς την αναδιατύπωση έχουμε επίσης $\langle \mathcal{Q}_h\lambda, \mathcal{Q}'' \rangle \in \mathcal{H}$ και $m_{\mathcal{Q}''} = m_{\mathcal{Q}_h\lambda}$. Τέλος, το $\bigwedge \mathsf{bd}(\mathcal{Q}'') \cup \mathsf{bd}(\mathcal{Q}_\alpha m_{\mathcal{Q}''}) \rightarrow \mathcal{Q}_A(nv')$ για $nv' = \mathsf{avar}(\mathcal{Q}'') \cup (\mathsf{avar}(\mathcal{Q}) \cap \mathsf{var}(\mathcal{Q}_\alpha)m_{\mathcal{Q}''}))$ είναι το $\mathcal{Q}_f$ και υπάγει το $\mathcal{Q}_{i+1}$, συνεπώς η Ιδιότητα $(\star)$ ικανοποιείται.

Έστω τώρα ότι για κάποιο $k \leq n$, το $\mathcal{Q}_\alpha\kappa$ περιέχει το άτομο $P(z_k, u)$ και ένα βήμα μείωσης περιλαμβάνει το $\mathcal{Q}_\alpha\kappa$ και τα άτομο από το $\mathcal{Q}_h$. Στην περίπτωση αυτή εφόσον το $\mathcal{Q}_\alpha\kappa$ περιέχει ένα άτομο η μείωση αυτή στην πραγματικότητα εξαλείφει το άτομο αυτό από το $\mathcal{Q}^+$ και συνεπώς το $\mathcal{Q}^+\lambda_k$ είναι της μορφής $\mathsf{bd}(\mathcal{Q}_h\lambda_k) \rightarrow \mathcal{Q}_A(\mathsf{avar}(\mathcal{Q})\lambda_k)$. Συνεπώς, εφαρμόζοντας τις υπόλοιπες μειώσεις το τελικό ΣΕ $\mathcal{Q}^+\lambda$ θα είναι της μορφής $\mathsf{bd}(\mathcal{Q}_h\lambda) \rightarrow \mathcal{Q}_A(\mathsf{avar}(\mathcal{Q})\lambda)$ και το $\mathcal{Q}_f$ μπορεί να παραχθεί εφαρμόζοντας το I σε αυτό το ΣΕ. Εφόσον τώρα, ο $\mathcal{G}$ είναι κλειστός ως προς την αναδιατύπωση όλες οι μειώσεις εκτός από αυτή με την λ_k έχουν εφαρμοστεί σε ερωτήματα στον $\mathcal{G}$ ξεκινώντας από το $\mathcal{Q}_h$, συνεπώς υπάρχει ένα ΣΕ $\mathcal{Q}_h\nu$, με $\nu = \lambda \setminus \lambda_k$ στον $\mathcal{G}$ τ.ω. $\mathcal{Q}_h \rightsquigarrow_{\mathcal{G}} \mathcal{Q}_h\nu$. Επιπλέον, εφόσον το $\mathcal{Q}_h\nu$ δεν περιέχει το άτομο που υπάρχει στο $\mathcal{Q}_\alpha\kappa$ το I είναι εφαρμόσιμο σε αυτό και συνεπώς $\langle \mathcal{Q}_h\nu, \mathcal{Q}'' \rangle \in \mathcal{H}$. Τότε, από τον ορισμό της συνάρτησης $\mathsf{mergeCQs}$, έχουμε ότι η αντικατάσταση $\mathsf{mergeCQs}(\mathcal{Q}_\alpha\nu, \mathcal{Q}_h\nu)$ περιέχει την $\{\lambda_k\}$. Άρα, το ΣΕ $\mathcal{Q}''\lambda_k$ είναι στην πραγματικότητα το $\mathcal{Q}_f$ που υπάγει το $\mathcal{Q}_{i+1}$ οπότε η Ιδιότητα $(\star)$ ικανοποιείται και πάλι.

2. Έστω ότι το $\mathcal{Q}^m$, δηλαδή το $[\bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv)]\mu'$ όπου $\mathcal{Q}_h \rightsquigarrow_{\mathcal{G}} \mathcal{Q}'$, με $\mu' = \mathsf{buildSubst}(\sigma, m(\mathcal{Q}'))$ και $\sigma \in \mathsf{mergeCQs}(\mathcal{Q}_\alpha\kappa, \mathcal{Q}_h)$ υπάγει το $\mathcal{Q}_j$. Και πάλι, από το Λήμμα 4.2.3 έχουμε ότι είτε το I είναι εφαρμόσιμο στο $\mathcal{Q}^m$ και το αποτέλεσμα (έστω $\mathcal{Q}_f$) υπάγει το $\mathcal{Q}_{i+1}$ ή επιπλέον περιορισμένα βήματα μείωσης είναι εφαρμόσιμα στο $\mathcal{Q}^m$ παράγοντας ένα ΣΕ στο οποίο το I είναι εφαρμόσιμο και το αποτέλεσμα (έστω και πάλι $\mathcal{Q}_f$) υπάγει το $\mathcal{Q}_{i+1}$. Εφόσον $\mathcal{Q}'m(\mathcal{Q}') = \mathcal{Q}'$ και εφόσον η μ' κατασκευάζεται από την σ και την $m(\mathcal{Q}')$, το $[\bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv)]\mu'$ είναι όπως το $\mathcal{Q}'$ αλλά πιθανότατα με κάποια μεταβλητή $z \in \mathsf{var}(\mathcal{Q}')$ που έχει μετονομαστεί λόγω κάποιας αντιστοίχισης $z \mapsto y \in \sigma$. Επιπλέον,

εφόσον το $[\bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv)]\mu'$ δεν αναφέρει την z (έχει μετονομαστεί λόγω της μ'), τότε κανένα από τα παραπάνω βήματα (εφαρμογή του I ή περιορισμένα βήματα μείωσης) δεν αφορούν την z. Τώρα, λόγω της μ' το $\mathcal{Q}'$ έχει λιγότερες δεσμευμένες μεταβλητές απο ότι το $[\bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv)]\mu'$ ή η μ' ελευθερώνει μια δεσμευμένη μεταβλητή που δεν είναι ελεύθερη στο $\mathcal{Q}'$ κάνοντας έτσι κάποιο αξίωμα εφαρμόσιμο. Στην πρώτη περίπτωση (το $\mathcal{Q}'$ έχει λιγότερες δεσμευμένες), η ίδια ακολουθία βημάτων αναδιατύπωσης ή περιορισμένων βημάτων μείωσης όπως στο $\mathcal{Q}^m$ είναι τουλάχιστον εφαρμόσιμα στο $\mathcal{Q}'$. Εφόσον αυτά δεν αφορούν τη μεταβλητή z, ύστερα από την εφαρμογή τους, ένα ΣΕ $\mathcal{Q}''$ τ.ω. $[\bigwedge \mathsf{bd}(\mathcal{Q}'') \rightarrow \mathcal{Q}_A(nv)]\mu'' = \mathcal{Q}_f$, όπου παράγεται η $\mu'' = \mathsf{buildSubst}(\sigma, m(\mathcal{Q}''))$, καθώς η μ'' που παράγεται από την σ θα εκτελέσει τις ίδιες μετονομασίες μεταβλητών στην z. Επιπλέον, εφόσον ο $\mathcal{G}$ είναι κλειστός ως προς την αναδιατύπωση το $\mathcal{Q}''$ ανήκει στον $\mathcal{G}$. Θεωρούμε τώρα την δεύτερη περίπτωση, ότι δηλαδή, η μ' ελευθερώνει μια δεσμευμένη μεταβλητή στο $\mathcal{Q}'$. Κάτι τέτοιο συνεπάγεται ότι η μ' επάγει μια αντιστοίχιση $\{z \mapsto x\}$ που ελευθερώνει κάποια δεσμευμένη μεταβλητή στο $\mathcal{Q}'$. Συνεπώς, ένα περιορισμένο βήμα μείωσης με ενοποιητή την $\sigma' = \{z \mapsto x\}$ είναι εφαρμόσιμο και εφόσον ο $\mathcal{G}$ είναι κλειστός ως προς την αναδιατύπωση θα έχουμε ότι $\langle \mathcal{Q}', \mathcal{Q}'\sigma' \rangle \in \mathcal{H}$. Στη συνέχεια, η ίδια ακολουθία βημάτω όπως στο $\mathcal{Q}^m$ είναι εφαρμόσιμα στο $\mathcal{Q}'\sigma'$. Άρα και πάλι, έχουμε ένα ερώτημα $\mathcal{Q}''$ τέτοιο ώστε $[\bigwedge \mathsf{bd}(\mathcal{Q}'') \rightarrow \mathcal{Q}_A(nv)]\mu'' = \mathcal{Q}_f$, όπου η $\mu'' = \mathsf{buildSubst}(\sigma, m(\mathcal{Q}''))$ παράγεται στον $\mathcal{G}$. Σε κάθε περίπτωση η Ιδιότητα $(\star)$ ικανοποιείται. $\square$

Κεφάλαιο 5

Επαυξητική Επανεγγραφή Ερωτημάτων

Στο προηγούμενο κεφάλαιο δείξαμε πώς μπορούμε, έχοντας μια DL-Lite$_R$ οντολογία, ένα ΣΕ, μια επανεγγραφή ΣΣΕ για το ερώτημα αυτό και ένα νέο άτομο, να κατασκευάσουμε μια επανεγγραφή ΣΣΕ για το νέο άτομο και στη συνέχεια να την ενώσουμε με αυτή του αρχικού ερώτηματος έτσι ώστε να παράγουμε μια επανεγγραφή ΣΣΕ για το ερώτημα που προκύπτει με την ένωση του αρχικού ερωτήματος και του ατόμου. Η διαδικασία αυτή υποδηλώνει ότι μπορεί να κατασκευαστεί ένας επαυξητικός αλγόριθμος επανεγγραφής ερωτημάτων για DL-Lite$_R$-TBox εάν για κάθε άτομο του ερωτήματος κατασκευαστεί μια επανεγγραφή και αυτές στη συνέχεια συνδυαστούν με κατάλληλο τρόπο. Ένα τέτοιο αλγόριθμο παρουσιάζουμε στην ενότητα 5.1. Στην συνέχεια, στην ενότητα 5.2 παρουσιάζουμε μια σειρά από βελτιστοποιήσεις που βοηθούν τον αλγόριθμο μας να παράγει μια επανεγγραφή ΣΣΕ η οποία περιέχει όσο το δυνατόν λιγότερα περιττά ερωτήματα. Πιο συγκεκριμένα μελετάμε βελτιστοποιήσεις που αφορούν στην προσθήκη του τελευταίου ατόμου του ερωτήματος (ενότητα 5.2.1) καθώς και βελτιστοποιήσεις οι οποίες μας βοηθάνε να αναγνωρίσουμε ερωτήματα που θα είναι (μη-)περιττά στην τελική επανεγγραφή αλλά και ερωτήματα που δεν πρόκειται να υπάγουν κανένα ερώτημα από την τελική επανεγγραφή (ενότητα 5.2.2). Οι βελτιστοποιήσεις αυτές, όπως θα φανεί και στο επόμενο κεφάλαιο, εκτός από το ότι συντελούν στο να παραχθεί μια επανεγγραφή ΣΣΕ με ελάχιστα περιττά ερωτήματα, αυξάνουν σημαντικά την αποδοτικότητα του αλγορίθμου μας.

5.1 Αλγόριθμος

Ο Αλγόριθμος 11 μπορεί να αποτελέσει τη βάση για την ανάπτυξη ενός αλγορίθμου παραγωγής μιας επανεγγραφής ΣΣΕ για *προκαθορισμένα* ερωτήματα. Πιο συγκεκριμένα, δεδομένου ενός ΣΕ $\mathcal{Q}$, κάποιος μπορεί να επιλέξει ένα άτομο $\alpha \in \mathcal{Q}$, να υπολογίσει τον γράφο επανεγγραφής για το ερώτημα $\mathcal{Q}_1 := \alpha \rightarrow \mathcal{Q}_A(\mathsf{var}(\alpha) \cap \mathsf{avar}(\mathcal{Q}))$ και το $\mathcal{T}$ χρησιμοποιώντας τον Αλγόριθμο ex-PerfectRef, και στη συνέχεια να τον επεκτείνει προσθέτοντας επαναληπτικά τα υπόλοιπα άτομα του ερωτήματος $\mathcal{Q}$ χρησιμοποιώντας τον Αλγόριθμο 11.

Αλγόριθμος 13 IncrementalRew$(\mathcal{Q}, \mathcal{T})$

Είσοδος: Ένα ΣΕ $\mathcal{Q}$ και ένα DL-Lite$_R$-TBox $\mathcal{T}$.

1: Έστω S το σύνολο από τα άτομα του σώματος του $\mathcal{Q}$
2: Αφαίρεσε ένα άτομο α από το S τ.ω. $\mathsf{var}(\alpha) \cap \mathsf{avar}(\mathcal{Q}) \neq \emptyset$
3: $\mathcal{G} := \mathsf{ex\text{-}PerfectRef}(\alpha \rightarrow \mathcal{Q}_A(\mathsf{var}(\alpha) \cap \mathsf{avar}(\mathcal{Q})), \mathcal{T})$
4: $cv := \mathsf{var}(\alpha)$
5: **while** $S \neq \emptyset$ **do**
6: Αφαίρεσε ένα άτομο α' από το S τ.ω. $\mathsf{var}(\alpha') \cap cv \neq \emptyset$
7: $jv := cv \cap \mathsf{var}(\alpha')$
8: $\mathcal{G}_{\alpha'} := \mathsf{ex\text{-}PerfectRef}(\alpha' \rightarrow \mathcal{Q}_A(jv), \mathcal{T})$
9: $\mathcal{G} := \mathsf{joinGraphs}(\mathcal{G}, \mathcal{G}_{\alpha'}, jv, \mathsf{avar}(\mathcal{Q}))$
10: $cv := cv \cup \mathsf{var}(\alpha')$
11: **end while**
12: **return** $\mathsf{removeRedundant}(\mathcal{G})$

Η παραπάνω ιδέα φαίνεται στον Αλγόριθμο 13. Ο επαυξητικός αυτός αλγόριθμος επιλέγει πρώτα κάποιο άτομο α τέτοιο ώστε κάποιες από τις μεταβλητές του να εμφανίζονται ως διακεκριμένες μεταβλητές στο ερώτημα $\mathcal{Q}$ (γραμμή 2) και υπολογίζει έναν γράφο επανεγγραφής $\mathcal{G}$ για το ερώτημα $\alpha \rightarrow \mathcal{Q}_A(\mathsf{var}(\alpha) \cap \mathsf{avar}(\mathcal{Q}))$ και το $\mathcal{T}$ (γραμμή 3). Στο σημείο αυτό έχει κατασκευαστεί ένας γράφος επανεγγραφής για ένα ερώτημα που περιέχει μόνο το άτομο α του $\mathcal{Q}$, τις μεταβλητές $cv := \mathsf{var}(\alpha)$ του $\mathcal{Q}$ και τις διακεκριμένες μεταβλητές $av := \mathsf{var}(\alpha) \cap \mathsf{avar}(\mathcal{Q})$ του $\mathcal{Q}$. Στη συνέχεια, ο αλγόριθμος επιλέγει ένα-ένα τα υπόλοιπα άτομα και επεκτείνει τον προηγούμενα υπολογισμένο γράφο επανεγγραφής (γραμμές 5–11). Πιο συγκεκριμένα, στην i-οστή επανάληψη ο αλγόριθμος έχει υπολογίσει έναν γράφο επανεγγραφής $\mathcal{G}$ για ένα ερώτημα $\mathcal{Q}_i$ (και το $\mathcal{T}$) που περιέχει $i+1$ άτομα του $\mathcal{Q}$, έχει ως διακεκριμένες μεταβλητές τις $\mathsf{var}(\mathcal{Q}_i) \cap \mathsf{avar}(\mathcal{Q})$, ενώ το cv περιέχει τις μεταβλητές του $\mathcal{Q}$ που εμφανίζονται στο $\mathcal{Q}_i$. Στη συνέχεια, ο αλγόριθμος επιλέγει ένα άτομο α' έτσι ώστε κάποιες από τις μεταβλητές του να εμφανίζονται στο cv (γραμμή 6), υπολογίζει έναν γράφο επανεγγραφής $\mathcal{G}_{\alpha'}$ για το ερώτημα $\alpha' \rightarrow \mathcal{Q}_A(\mathsf{var}(\alpha') \cap cv)$ και το $\mathcal{T}$ (γραμμή 8) και στη συνέχεια, ενώνει τον $\mathcal{G}$ με τον $\mathcal{G}_{\alpha'}$ χρησιμοποιώντας τον αλγόριθμο 11 (γραμμή 9). Τέλος, ο αλγόριθμος χρησιμοποιεί τον ευρέως γνωστό αλγόριθμο απαλοιφής περιττών ερωτημάτων removeRedundant (Αλγόριθμος 1) έτσι ώστε να αφαιρέσει όλα τα περιττά ερωτήματα (γραμμή 12). Αξίζει να σημειωθεί ότι η απαλοιφή των περιττών ερωτημάτων γίνεται εφόσον το ερώτημα είναι προκαθορισμένο και γνωρίζουμε ότι δεν πρόκειται να επεκταθεί περισσότερο.

Παράδειγμα 5.1.1. Έστω το TBox $\mathcal{T}$ από το Παράδειγμα 3.4.1:

$$\begin{aligned} \mathsf{Καθηγητής}(x) &\rightarrow \exists y.\mathsf{διδάσκει}(x, y), \\ \exists y.\mathsf{διδάσκει}(y, x) &\rightarrow \mathsf{Φοιτητής}(x) \end{aligned}$$

και το ΣΕ

$$\mathcal{Q} = \mathsf{διδάσκει}(x, y) \wedge \mathsf{Φοιτητής}(y) \rightarrow \mathcal{Q}_A(x)$$

που ανακτά όλα τα άτομα που διδάσκουν κάποιον φοιτητή. Ο Αλγόριθμος 13 ξεκινά την εκτέλεση του επιλέγοντας το άτομο $\alpha = \text{διδάσκει}(x, y)$ από το σύνολο $S = \{\text{διδάσκει}(x, y), \text{Φοιτητής}(y)\}$, αφού είναι το μόνο για το οποίο ισχύει $\text{var}(\alpha) \cap \text{avar}(q) \neq \emptyset$ (γραμμή 2). Στη συνέχεια υπολογίζει το γράφο επανεγγραφής των $\mathcal{Q}_\alpha = \text{διδάσκει}(x, y) \rightarrow \mathcal{Q}_A(x), \mathcal{T}$ που είναι ο $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$ (γραμμή 3) με $\mathcal{R}_Q = \{\mathcal{Q}_\alpha, \mathcal{Q}_{1\alpha}\}$, όπου $\mathcal{Q}_{1\alpha} = \text{Καθηγητής}(x) \rightarrow \mathcal{Q}_A(x)$, $\mathcal{H} = \{\langle \mathcal{Q}_\alpha, \mathcal{Q}_{1\alpha} \rangle\}$ και $m(\mathcal{Q}_\alpha) = m(\mathcal{Q}_{1\alpha}) = \emptyset$, ενώ θέτει και το σύνολο των μεταβλητών $cv = \{x, y\}$ (γραμμή 4). Συνεχίζοντας ο αλγόριθμος εισέρχεται στον βρόχο επανάληψης (γραμμές 5–11). Εκεί αφαιρεί το άτομο $\alpha' = \text{Φοιτητής}(y)$ για το οποίο ισχύει $\text{var}(\alpha') \cap cv \neq \emptyset$ και υπολογίζει το σύνολ $jv = \{y\}$. Στη συνέχεια υπολογίζει το γράφο επανεγγραφής των $\mathcal{Q}_{\alpha'} = \text{Φοιτητής}(y) \rightarrow \mathcal{Q}_A(y), \mathcal{T}$ που δίνεται από τον $\mathcal{G}_{\alpha'} = \langle \mathcal{R}_{\alpha'}, \mathcal{H}_{\alpha'}, m_{\alpha'} \rangle$ (γραμμή 8) με $\mathcal{R}'_\alpha = \{\mathcal{Q}_{\alpha'}, \mathcal{Q}_{1\alpha'}\}$, όπου $\mathcal{Q}_{1\alpha'} = \text{διδάσκει}(z, y) \rightarrow \mathcal{Q}_A(y)$ για μια νέα μεταβλητή z, $\mathcal{H}_{\alpha'} = \{\langle \mathcal{Q}_{\alpha'}, \mathcal{Q}_{1\alpha'} \rangle\}$ και $m(\mathcal{Q}_{\alpha'}) = m(\mathcal{Q}_{1\alpha'}) = \emptyset$. Στο σημείο αυτό ο αλγόριθμος υπολογίζει τον γράφο επανεγγραφής που υπολογίζεται από τη συνάρτηση joinGraphs με ορίσματα τα $\mathcal{G}$, $\mathcal{G}_{\alpha'}$, $\{y\}$ και $\{x\}$ (γραμμή 9). Ο γράφος αυτός είναι ο $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$ με $\mathcal{R}_Q = \{\mathcal{Q}, \mathcal{Q}'_1, \mathcal{Q}_\alpha, \mathcal{Q}_{1\alpha}\}$, όπου $\mathcal{Q}'_1 = \text{διδάσκει}(x, y) \wedge \text{διδάσκει}(z, y) \rightarrow \mathcal{Q}_A(x)$, $\mathcal{H} = \{\langle \mathcal{Q}, \mathcal{Q}'_1 \rangle, \langle \mathcal{Q}'_1, \mathcal{Q}_\alpha \rangle, \langle \mathcal{Q}_\alpha, \mathcal{Q}_{1\alpha} \rangle\}$, και $m(\mathcal{Q}) = m(\mathcal{Q}'_1) = \emptyset$ και $m(\mathcal{Q}_\alpha) = m(\mathcal{Q}_{1\alpha}) = \{z \mapsto x\}$. Τέλος, στη γραμμή 12 γίνεται ο έλεγχος απαλοιφής περιττών ερωτημάτων για τον γράφο G που έχει ως αποτέλεσμα την απαλοιφή των ερωτημάτων $\mathcal{Q}$ και $\mathcal{Q}'_1$ και την επιστροφή της επανεγγραφής ΣΣΕ $\{\mathcal{Q}_\alpha, \mathcal{Q}_{1\alpha}\}$ για τα $\mathcal{Q}, \mathcal{T}$.

Θεώρημα 5.1.2. *Έστω $\mathcal{Q}$ ΣΕ και έστω $\mathcal{T}$ DL-Lite$_R$-TBox. Όταν εφαρμόζεται στα $\mathcal{Q}$ και $\mathcal{T}$ ο Αλγόριθμος 13 τερματίζει. Έστω $\mathcal{G}$ γράφος που παράγεται από τον αλγόριθμο. Τότε, ο $\mathcal{G}$ είναι ένας γράφος επανεγγραφής για τα $\mathcal{Q}, \mathcal{T}$.*

Απόδειξη: Για να δείξουμε το θεώρημα αυτό χρησιμοποιήσουμε επαγωγή στα άτομα του $\mathcal{Q}$ ($\{\alpha_1, \ldots, \alpha_n\}$) που επεξεργάζεται ο Αλγόριθμος 13.

Βασική Περίπτωση ($\alpha = \alpha_1$): Όταν ο αλγόριθμος επεξεργάζεται το πρώτο άτομο α_1 για το οποίο $\text{var}(\alpha_1) \cap \text{avar}(\mathcal{Q}) \neq \emptyset$ παράγει ένα γράφο επανεγγραφής $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$ για το ερώτημα $\mathcal{Q}_{\alpha_1} = \alpha_1 \rightarrow \mathcal{Q}_A(\text{var}(\alpha_1) \cap \text{avar}(\mathcal{Q})))$ και το $\mathcal{T}$ με χρήση του αλγορίθμου ex-PerfectRef (γραμμή 3). Ο αλγόριθμος αυτός έχει αποδειχθεί ότι τερματίζει και ο γράφος που παράγει είναι κλειστός ως προς την αναδιατύπωση εφόσον ο ex-PerfectRef εφαρμόζει εξαντλητικά το βήμα αναδιατύπωσης (το βήμα μείωσης δεν εφαρμόζεται εφόσον το $\mathcal{Q}_{\alpha_1}$ έχει μόνο ένα άτομο στο σώμα του) και δεν κλαδεύει τα υπολογισμένα ερωτήματα.

Βήμα Επαγωγής: Υποθέτουμε ότι ο γράφος επανεγγραφής $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$ που παράγεται όταν προστίθεται το άτομο α_i είναι κλειστός ως προς την αναδιατύπωση. Με βάση το Θεώρημα 4.2.5 όταν προστεθεί και το άτομο α_{i+1} για το οποίο θα παραχθεί ο γράφος επανεγγραφής $\mathcal{G}_{\alpha'}$ η συνάρτηση joinGraphs η οποία καλείται με ορίσματα τα $\mathcal{G}$, $\mathcal{G}_{\alpha'}$, jv και av θα παράγει ένα γράφο επανεγγραφής που είναι κλειστός ως προς την αναδιστύπωση. Επιπλέον με βάση το Θεώρημα 4.2.5 η διαδικασία αυτή τερματίζει. □

Στο σημείο αυτό σχολιάζουμε τον μέγιστο αριθμό ΣΕ που παράγονται από τον αλγόριθμο επανεγγραφής μας. Εφόσον ο αλγόριθμος υπολογίζει μια επανεγγραφή ΣΣΕ ο μέγιστος αριθμός των ΣΕ παρουσιάζει την ίδια πολυπλοκότητα στη χειρότερη περίπτωση με άλλα συστήματα, όπως το QuOnto, το Nyaya, και το Requiem, δηλαδή,

εκθετική μ.β.τ. το μέγεθος του ΣΕ που δίνεται ως είσοδος [85].

Λήμμα 5.1.3. *Έστω $\mathcal{T}$ DL-Lite TBox, έστω $\mathcal{Q}$ ΣΕ, και έστω $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$ έξοδος του Αλγόριθμου 13. Τότε, το μέγιστο μέγεθος του $|\mathcal{R}_Q|$ είναι $\mathcal{O}((|\mathcal{T}| \cdot |\mathcal{Q}|)^{|\mathcal{Q}|})$.*

Απόδειξη: Έστω m ο αριθμός των εννοιών και ρόλων που εμφανίζονται στο $\mathcal{T}$ και n ο αριθμός των ατόμων στο $\mathcal{Q}$. Έστω επίσης $\mathcal{R}_i$ το ΣΣΕ που υπολογίζεται κατά την i-1-στή επανάληψη του βρόχου while του Αλγορίθμου 13. Στην επόμενη επανάληψη, το $\mathcal{R}_{i+1}$ υπολογίζεται κατασκευάζοντας πρώτα ένα γράφο επανεγγραφής $\mathcal{G}_{\alpha_i} = \langle \mathcal{R}_{\alpha_i}, \mathcal{H}_{\alpha_i}, m_{\alpha_i} \rangle$ για ένα άτομο α_i του $\mathcal{Q}$ και στη συνέχεια χρησιμοποιώντας τα ερωτήματα του $\mathcal{R}_{\alpha_i}$ και του $\mathcal{R}_i$ για την κατασκευή νέων ερωτημάτων είτε επεκτείνοντας, είτε συγχωνεύοντας. Εφόσον το ερώτημα για το οποίο υπολογίζεται το $\mathcal{R}_{\alpha_i}$ περιέχει ακριβώς ένα άτομο, τότε υπάρχουν το πολύ $m \cdot (2+1)^2$ διαφορετικά ερωτήματα στο $\mathcal{R}_\alpha$ (ο παράγοντας '$2+1$' προκύπτει επειδή το άτομο μπορεί να είναι ρόλος οπότε υπάρχουν δύο μεταβλητές συν 1 για πιθανές νέες μεταβλητές). Στη συνέχεια, ο αλγόριθμος επεκτείνει τα ερωτήματα στο $\mathcal{R}_i$ με άτομα από τα ερωτήματα στο $\mathcal{R}_{\alpha_i}$. Συνεπώς, μπορεί να παραχθούν $|\mathcal{R}_i| \cdot m \cdot 3^2$ ΣΕ από επέκταση. Επιπλέον, ο αλγόριθμος ελέγχει εάν ερωτήματα από το $\mathcal{R}_\alpha$ μπορούν να συγχωνευθούν με ΣΕ υπολογίζοντας όλες τις δυνατές συγχωνεύσεις. Για κάθε συγχώνευση, όλα τα ΣΕ στο $\mathcal{R}_i$ που βρίσκονται 'κάτω' από το ερώτημα που συγχωνεύεται αντιγράφοντα. Από τη στιγμή που τα ερωτήματα στο $\mathcal{R}_i$ έχουν το πολύ i άτομα (μέχρι στιγμής έχουν επεξεργαστεί i άτομα), μπορούν να υπάρχουν το πολύ $i \cdot |\mathcal{R}_i| \cdot m \cdot 3^2$ ΣΕ που υπολογίζονται λόγω συγχώνευσης. Συνοψίζοντας, το $|\mathcal{R}_{i+1}|$ περιέχει το πολύ $(i+1) \cdot |\mathcal{R}_i| \cdot m \cdot 3^2$ ΣΕ. Αναλύοντας αναδρομικά το $|\mathcal{R}_i|$ και εφόσον $|\mathcal{R}_0| = \mathcal{R}_{\alpha_0}$ μπορούμε να συμπεράνουμε ότι το $|\mathcal{R}_n|$ περιέχει το πολύ $n \cdot (n-1) \cdot \ldots \cdot 2 \cdot m^n \cdot 3^{2 \cdot n}$ ΣΕ. Εφόσον ο m είναι δεσμευμένος από τον $|\mathcal{T}|$ και ο n από τον $|\mathcal{Q}|$ έχουμε ότι $|\mathcal{R}_Q| = |\mathcal{R}_n| = \mathcal{O}(|\mathcal{T}|^{|\mathcal{Q}|} \cdot |\mathcal{Q}|! \cdot 3^{2 \cdot |\mathcal{Q}|}) = \mathcal{O}(|\mathcal{T}|^{|\mathcal{Q}|} \cdot |\mathcal{Q}|!) = \mathcal{O}((|\mathcal{T}| \cdot |\mathcal{Q}|)^{|\mathcal{Q}|})$. □

5.2 Βελτιστοποιήσεις

Ένας πολύ σημαντικός παράγοντας για την επανεγγραφή ερωτημάτων είναι να μπορεί να υπολογιστεί όσο το δυνατόν πιο αποδοτικά μια επανεγγραφή και αυτό στις πλείστες των περιπτώσεων σημαίνει ότι το μέγεθος της επανεγγραφής πρέπει να περιέχει όσο το δυνατόν λιγότερα περιττά ερωτήματα. Στις ενότητες που ακολουθούν παρουσιάζουμε διάφορες βελτιστοποιήσεις που έχουν ως στόχο την βελτίωση της απόδοσης του Αλγόριθμου 13. Οι βελτιστοποιήσεις αυτές εχουν ως στόχο να βελτιώσουν την επίδοση του, δηλαδή τον χρόνο που χρειάζεται ώστε να παράγει μια επανεγγραφή. Θυμίζουμε ότι υπάρχουν και τεχνικές που έχουν αναπτυχθεί πρόσφατα οι οποίες δείχνουν πως μπορούμε να μειώσουμε επίσης και το μέγεθος του ΣΣΕ χρησιμοποιώντας τα δεδομένα εισόδου όταν θέλουμε τελικά να αποτιμήσουμε την επανεγγραφή [127, 131].

5.2.1 Βελτιστοποιώντας το Τελευταίο Βήμα

Όπως έχει εξηγηθεί στο Κεφάλαιο 4, ο Αλγόρθμος 11 υπολογίζει το καρτεσιανό γινόμενο μεταξύ των γράφων που δίνονται ως είσοδος. Ενώ η δομή του γράφου που υπολογίζεται είναι σημαντική για την περαιτέρω προσθήκη ατόμων δεν είναι σημαντική μετά την επεξεργασία του τελευταίου ατόμου ενός *προκαθορισμένου* ερωτήματος χρησιμοποιώντας τον Αλγόριθμο 13. Συνεπώς, όταν ο Αλγόριθμος 13 επιλέγει το τελευταίο άτομο του ερωτήματος εισόδου α' (γραμμή 6) μπορεί να ενεργήσει ως εξής: Πρώτα, στη γραμμή 8 μπορεί να υπολογίσει μια επανεγγραφή ΣΣΕ $\mathcal{R}_{\alpha'}$ για το ερώτημα $\alpha' \rightarrow \mathcal{Q}_A(jv)$, χρησιμοποιώντας για παράδειγμα τον αλγόριθμο PerfectRef, αντί για έναν γράφο επανεγγραφής $\mathcal{G}_{\alpha'}$. Τότε, όταν ενώνει τον $\mathcal{G}$ και το $\mathcal{R}_{\alpha'}$ μπορεί να καλέσει μια απλοποιημένη έκδοση του Αλγόριθμου 11 που παράγει ένα ΣΣΕ αντί για έναν γράφο επανεγγραφής. Ο Αλγόριθμος 14 απεικονίζει τον απλοποιημένο αυτό αλγόριθμο. Σε γενικές γραμμές, ο αλγόριθμος αυτός προκύπτει από τον Αλγόριθμο 11 αφαιρώντας τον βρόχο επανάληψης στις γραμμές 27—30 και προσθέτοντας τα ερωτήματα που παράγονται σε ένα ΣΣΕ $\mathcal{R}'_Q$ αντί σε έναν γράφο.

Ένας άλλος τρόπος να βελτιώσουμε επιπλέον την αποδοτικότητα του Αλγόριθμου 14 είναι να αναγνωρίσουμε περιπτώσεις που τα ερωτήματα από τον $\mathcal{G}$ δεν χρειάζεται να επεκταθούν περαιτέρω με άτομα των ερωτημάτων του $\mathcal{R}_\alpha$. Η επόμενη πρόταση δείχνει ότι όταν ένα ερώτημα $\mathcal{Q}'$ τ.ω. $\langle \mathcal{Q}, \mathcal{Q}' \rangle \in \mathcal{G}$ «αντιγράφεται» στο $\mathcal{R}'_Q$ λόγω της συγχώνευσης του $\mathcal{Q}$ με το $\mathcal{Q}_\alpha$, τότε το $\mathcal{Q}$ δεν χρειάζεται να επεξεργαστεί περαιτέρω από τον αλγόριθμο.

Πρόταση 5.2.1. *Έστω $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$ γράφος επανεγγραφής, έστω $\mathcal{R}_\alpha$ ΣΣΕ, έστω nv σύνολο από μεταβλητές και έστω $\mathcal{Q}_h \in \mathcal{G}$. Εάν υπάρχει ερώτημα $\mathcal{Q}_\alpha$ στο $\mathcal{R}_\alpha$ και αντικατάσταση λ τέτοια ώστε για κάθε $\{z \mapsto x\} \in \mathsf{mergeCQs}(\mathcal{Q}_\alpha\lambda, \mathcal{Q})$ και $z \notin \mathsf{var}(\mathcal{Q}_h)$ έχουμε $z \notin \mathsf{var}(\mathcal{Q}_h)$, τότε για κάθε ερώτημα $\mathcal{Q}'$ τέτοιο ώστε $\mathcal{Q}_h \rightsquigarrow_{\mathcal{G}} \mathcal{Q}'$, κάθε $\mathcal{Q}'_\alpha$ στο $\mathcal{R}_\alpha$ και κάθε αντικατάσταση κ, το ΣΕ $[\bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv)]\mu'$ όπου $\mu' = \mathsf{buildSubst}(\{z \mapsto x\}, m(\mathcal{Q}'))$ υπάγει το $\bigwedge \mathsf{bd}(\mathcal{Q}') \cup \mathsf{bd}(\mathcal{Q}'_\alpha\kappa) \rightarrow \mathcal{Q}_A(nv)$.*

Απόδειξη: Έστω $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$ γράφος επανεγγραφής, $\mathcal{R}_\alpha$ επανεγγραφή ΣΣΕ, και jv και av σύνολα από μεταβλητές τα οποία δίνονται ως είσοδος στον Αλγόριθμο 14. Υποθέτουμε ότι για κάποιο ερώτημα $\mathcal{Q}_h$ που έχει επιλεγεί στην γραμμή 4, κάποιο ερώτημα $\mathcal{Q}_\alpha$ που έχει επιλεγεί στην γραμμή 6 και κάποια $\sigma = \{z \mapsto x\} \in \mathsf{mergeCQs}(\mathcal{Q}_\alpha\lambda, \mathcal{Q})$ στην γραμμή 9 ο αλγόριθμος προσθέτει όλα τα ΣΕ $[\bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv)]\mu'$ με $\mathcal{Q}_h \rightsquigarrow_{\mathcal{G}} \mathcal{Q}'$ στο $\mathcal{R}'_Q$ (γραμμή 14). Εάν $z \notin \mathsf{var}(\mathcal{Q})$, τότε από τις ιδιότητες της αναδιατύπωσης και της μείωσης ισχύει επίσης $z \notin \mathsf{var}(q')$ για κάθε τέτοιο ερώτημα $\mathcal{Q}'$ και συνεπώς καμία αντιστοίχιση της μορφής $z \mapsto y$ δεν εμφανίζεται στην $m(\mathcal{Q}')$. Επιπλέον, έχουμε $\mathcal{Q}'m(\mathcal{Q}') = \mathcal{Q}'$. Συνεπώς, για κάθε τέτοιο ερώτημα $\mathcal{Q}'$ και για $\mu' = \mathsf{buildSubst}(\{z \mapsto x\}, m(\mathcal{Q}'))$ έχουμε $\mathcal{Q}'\mu' = \mathcal{Q}'$ και συνεπώς το $[\bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv)]\mu'$ υπάγει το $\bigwedge \mathsf{bd}(\mathcal{Q}') \cup \mathsf{bd}(\mathcal{Q}'_\alpha\kappa) \rightarrow \mathcal{Q}_A(nv)$ για κάθε κ και $\mathcal{Q}'_\alpha$. □

Βλέπουμε λοιπόν, πως ο Αλγόριθμος 14 προσθέτει στην ουρά τα ερωτήματα που προκύπτουν από ένα επιλεγμένο ερώτημα $\mathcal{Q}_h$ στη γραμμή 19 και τα επεξεργάζεται περαιτέρω μόνο εάν $\Sigma = \emptyset$ ή υπάρχει $\{z \mapsto x\} \in \Sigma$ με $z \in \mathsf{var}(\mathcal{Q}_h)$. Διαφορετικά,

Αλγόριθμος 14 OptimisedExtensionStep$(\mathcal{G}, u_\alpha, jv, av)$

Είσοδος: Ένας γράφος επανεγγραφής $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$, μια επανεγγραφή ΣΣΕ $\mathcal{R}_\alpha$, και το σύνολο των μεταβλητών ένωσης και των διακεκριμένων μεταβλητών.

1: Αρχικοποίησε μια ουρά Q με ένα κόμβο κορυφής $\mathcal{G}$
2: Αρχικοποίησε ένα ΣΣΕ $\mathcal{R}'_Q := \emptyset$
3: **while** $Q \neq \emptyset$ **do**
4: Αφαίρεσε την κεφαλή $\mathcal{Q}_h$ της Q και έστω $\kappa := m(\mathcal{Q}_h)$
5: **if** canBeJoined$(\mathcal{Q}_h, \kappa, jv)$ **then**
6: **for all** $q_\alpha \in \mathcal{R}_\alpha$ **do**
7: $nv :=$ avar$(\mathcal{Q}_h) \cup$ (var$(\mathcal{Q}_\alpha\kappa \cap av)$)
8: Προσθεσε το $\mathcal{Q}_h \cup \mathcal{Q}_\alpha\kappa \rightarrow \mathcal{Q}_A(nv)$ στο $\mathcal{R}'_Q$
9: $\Sigma :=$ mergeCQs$(\mathcal{Q}_\alpha\kappa, \mathcal{Q}_h)$
10: **for all** $\sigma \in \Sigma$ **do**
11: **for all** $\mathcal{Q}'$ τ.ω. $\mathcal{Q}_h \rightsquigarrow_{\mathcal{G}} \mathcal{Q}'$ **do**
12: **if** dom$(\sigma) \subseteq$ var$(\mathcal{Q}') \cup$ dom$(m(\mathcal{Q}'))$ **then**
13: $\mu' :=$ buildSubst$(\sigma, m(\mathcal{Q}'))$
14: Πρόσθεσε το $[\mathcal{Q}' \rightarrow \mathcal{Q}_A(nv)]\mu'$ στο $\mathcal{R}'_Q$
15: **end if**
16: **end for**
17: **end for**
18: **if** $(\Sigma = \emptyset$ or $\{z \mapsto x\} \in \Sigma \wedge z \in$ var$(\mathcal{Q}))$ **then**
19: Πρόσθεσε κάθε $\mathcal{Q}'$ τ.ω. $\langle \mathcal{Q}_h, \mathcal{Q}' \rangle \in \mathcal{G}$ στην Q
20: **end if**
21: **end for**
22: **end if**
23: **end while**
24: **return** $\mathcal{R}'_Q$

όπως προκύπτει από την Πρόταση 5.2.1 οι επεκτάσεις τέτοιων ερωτημάτων οδηγούν στη δημιουργία ερωτημάτων που είναι περιττά.

5.2.2 Βελτιστοποιώντας την Απαλοιφή Περιττών Ερωτημάτων

Όπως έχουμε ήδη περιγράψει (ενότητα 5.1), ο Αλγόριθμος 13 στη γραμμή 12 εφαρμόζει τον γνωστό αλγόριθμο απαλοιφής περιττών ερωτημάτων removeRedundant που φαίνεται στον Αλγόριθμο 1. Όπως έχει δειχθεί από διάφορες πειραματικές αξιολογήσεις [116, 40], η μέθοδος αυτή συνήθως δεν αποδίδει καλά στην πράξη, επειδή αποτελείται από δύο εμφωλευμένους βρόχους επανάληψης πάνω σε μια (πιθανότατα μεγάλη) υπολογισμένη επανεγγραφή $\mathcal{R}'_Q$.[1] Πιο συγκεκριμένα, ο αλγόριθμος πρέπει να ελέγχει εάν κάθε ΣΕ $\mathcal{Q}_1 \in \mathcal{R}'_Q$ υπάγεται από κάποιο άλλο ΣΕ $\mathcal{Q}_2 \in \mathcal{R}'_Q$. Για να

[1] Θυμίζουμε ότι σε μια επανεγγραφή ΣΣΕ ισχύει $\mathcal{R}_D = \emptyset$.

βελτιώσουμε την απόδοση της μεθόδου αυτής ο αλγόριθμος μας χρησιμοποιεί τρεις προσεγγίσεις, όμοιες με αυτές που παρουσιάσαμε στην ενότητα 3.5 με τις οποίες προσπαθεί να μειώσει το μέγεθος των συνόλων τα οποία θα προσπελάσει ο αλγόριθμος στους βρόχους αυτούς.

Έτσι, λοιπόν, κατά τη διάρκεια του τρεξίματος του Αλγορίθμου 14 ο αλγόριθμος μας προσπαθεί να αναγνωρίσει ερωτήματα που εάν παραχθούν και προστεθούν στο σύνολο $\mathcal{R}'_Q$ πρόκειται να είναι περιττά, μειώνοντας όσο το δυνατόν περισσότερο το μέγεθος του συνόλου $\mathcal{R}'_Q$ για το οποίο θα πρέπει να εφαρμοστεί ο αλγόριθμος removeRedundant. Επειδή όμως το $\mathcal{R}'_Q$ μπορεί και πάλι να είναι πολύ μεγάλο, ο αλγόριθμος μας στη συνέχεια προσπαθεί να αναγνωρίσει ερωτήματα που πρόκειται να είναι μη-περιττά στο σύνολο $\mathcal{R}'_Q$ που υπολογίζεται, έτσι ώστε να τα αποκλείσει από τον τελικό έλεγχο, μειώνοντας το μέγεθος του πρώτου βρόχου της μεθόδου removeRedundant στο σύνολο $\mathcal{R}_Q \setminus NR$, όπου NR είναι το σύνολο των μη-περιττών ερωτημάτων. Τέλος, ο αλγόριθμος μας προσπαθεί να αναγνωρίσει ερωτήματα που είναι μη-υπάγοντα, έτσι ώστε να μπορέσει να μειώσει το μέγεθος του δεύτερου βρόχου στο σύνολο $\mathcal{R}_Q \setminus NS$, όπου το NS περιέχει όλα αυτά τα ερωτήματα Έχοντας αναγνωρίσει, λοιπόν, ο αλγόριθμος μας ερωτήματα που θα είναι μη-περιττά και μη-υπάγοντα στην τελική επανεγγραφή μπορεί στη συνέχεια αντί για τον Αλγόριθμο 1 για την απαλοιφή των περιττών ερωτημάτων να χρησιμοποιήσει τον βελτιστοποιημένο Αλγόριθμο 7.

Απαλοίφοντας Περιττά Ερωτήματα

Η παρακάτω πρόταση παρουσιάζει δύο ιδιότητες με την χρήση των οποίων μπορούμε να αναγνωρίσουμε εάν ένα ερώτημα που κατασκευάζεται κατά την εκτέλεση του Αλγορίθμου 14 θα είναι περιττό στο τελικό ΣΣΕ.

Πρόταση 5.2.2. *Έστω $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$ γράφος επανεγγραφής, έστω $\mathcal{R}_\alpha$ ΣΣΕ, έστω jv, av σύνολα από μεταβλητές, και έστω $\mathcal{R}'_Q$ έξοδος του Αλγόριθμου 14 όταν αυτός εφαρμόζεται στα $\mathcal{G}, \mathcal{R}_\alpha, jv$, και av. Έστω ΣΕ $\mathcal{Q} \in \mathcal{R}_Q$ τ.ω.* canBeJoined$(\mathcal{Q}, m(\mathcal{Q}), jv) =$ αληθές*, έστω nv σύνολο μεταβλητών που κατασκευάζεται για το $\mathcal{Q}$ στην γραμμή 7, και έστω πως το $\mathcal{Q}$ υπάγεται από κάποιο ΣΕ $\mathcal{Q}' \in \mathcal{R}_Q$. Τότε ισχύουν οι ακόλουθες ιδιότητες:*

(R1) *Εάν $\bigwedge$ bd$(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv) \in \mathcal{R}'_Q$, τότε το $\bigwedge$ bd$(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv)$ είναι περιττό στο $\mathcal{R}'_Q$.*

(R2) *Εάν* canBeJoined$(\mathcal{Q}', m(\mathcal{Q}'), jv) =$ αληθές*, $\mathcal{Q}' \subseteq \mathcal{Q}$, και $m(\mathcal{Q}') = m(\mathcal{Q})$, τότε για κάθε $\mathcal{Q}_\alpha \in \mathcal{R}_\alpha$ το ΣΕ $\bigwedge$ bd$(\mathcal{Q} \cup$ bd$(\mathcal{Q}_\alpha m(\mathcal{Q}))) \rightarrow \mathcal{Q}_A(nv)$ είναι περιττό στο $\mathcal{R}'_Q$.*

Απόδειξη: Η Ιδιότητα $(R1)$ προκύπτει ευθέως, οπότε δείχνουμε μόνο την Ιδιότητα $(R2)$.

Πρώτα δείχνουμε ότι για κάθε $\mathcal{Q}_\alpha \in \mathcal{R}_\alpha$ το ΣΕ $\bigwedge$ bd$(\mathcal{Q}) \cup$ bd$(\mathcal{Q}_\alpha m(\mathcal{Q})) \rightarrow \mathcal{Q}_A(nv)$ υπάγεται από το ΣΕ $\bigwedge$ bd$(\mathcal{Q}') \cup$ bd$(\mathcal{Q}_\alpha m(\mathcal{Q}')) \rightarrow \mathcal{Q}_A(nv')$ όπου το nv' είναι το σύνολο των μεταβλητών που κατασκευάζονται για το $\mathcal{Q}'$ στη γραμμή 7. Πιο συγκεκριμένα, εφόσον $\mathcal{Q}' \subseteq \mathcal{Q}$ και $m(\mathcal{Q}') = m(\mathcal{Q})$ για κάθε $\mathcal{Q}_\alpha \in \mathcal{R}_\alpha$ έχουμε $\bigwedge$ bd$(\mathcal{Q}') \cup$

$\mathsf{bd}(\mathcal{Q}_\alpha m(\mathcal{Q}')) \to \mathcal{Q}_A(\mathsf{avar}(\mathcal{Q}')) \subseteq \bigwedge \mathsf{bd}(\mathcal{Q}) \cup \mathsf{bd}(\mathcal{Q}_\alpha m(\mathcal{Q})) \to \mathcal{Q}_A(\mathsf{avar}(\mathcal{Q}))$. Επιπλέον, για τους ίδιους λόγους το nv είναι ίδιο με το nv' και άρα το $\bigwedge \mathsf{bd}(\mathcal{Q}') \cup \mathsf{bd}(\mathcal{Q}_\alpha m(\mathcal{Q}')) \to \mathcal{Q}_A(nv')$ υπάγει το $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \mathsf{bd}(\mathcal{Q}_\alpha m(\mathcal{Q})) \to \mathcal{Q}_A(nv)$ για κάθε $\mathcal{Q}_\alpha \in \mathcal{R}_\alpha$.

Τώρα, εφόσον $\mathsf{canBeJoined}(\mathcal{Q}', m(\mathcal{Q}'), jv) = \mathsf{αληθές}$, κατά τον τερματισμό του Αλγορίθμου 14, έχουμε ότι για κάθε $\mathcal{Q}_\alpha \in \mathcal{R}_\alpha$ είτε το ΣΕ $\bigwedge \mathsf{bd}(\mathcal{Q}') \cup \mathsf{bd}(\mathcal{Q}_\alpha m(\mathcal{Q}')) \to \mathcal{Q}_A(nv')$ έχει προστεθεί στο $\mathcal{R}'_Q$ στη γραμμή 8 ή ένα άλλο ερώτημα που το υπάγει. Σε κάθε περίπτωση το $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \mathsf{bd}(\mathcal{Q}_\alpha m(\mathcal{Q})) \to \mathcal{Q}_A(nv)$ είναι περιττό στο $\mathcal{R}'_Q$. □

Η Πρόταση 5.2.2 μπορεί να χρησιμοποιηθεί για την αποφυγή της προσθήκης περιττών ερωτημάτων στο αποτέλεσμα $\mathcal{R}'_Q$ όταν ο Αλγόριθμος 13 καλεί τον Αλγόριθμο 14 κατά την τελευταία επανάληψη. Παρόλα αυτά, για να μπορέσουμε να ελέγξουμε εάν ισχύουν οι Ιδιότητες $(R1)$ και $(R2)$ πρέπει να γνωρίζουμε τις σχέσεις υπαγωγής που ισχύουν ανάμεσα στα ΣΕ στον $\mathcal{G}$. Για να αναγνωρίσουμε τις σχέσεις αυτές, πριν καλέσουμε τον Αλγόριθμο 14, ο Αλγόριθμος 13 εκτελεί τον πρωτότυπο αλγόριθμο υπαγωγής (Αλγόριθμος 1) για τον $\mathcal{G}$ και αποθηκεύει όλες αυτές τις σχέσεις. Αξίζει να σημειωθεί ότι στο σημείο αυτό δεν αφαιρούνται ερωτήματα καθώς ο γράφος $\mathcal{G}$ πρέπει να είναι κλειστός ως προς την αναδιατύπωση έτσι ώστε ο Αλγόριθμος 14 να κατασκευάσει μια επανεγγραφή ΣΣΕ. Επιπρόσθετα, πρέπει να σημειωθεί ότι το μέγεθος του $\mathcal{G}$ στο σημείο αυτό αναμένεται να είναι σημαντικά μικρότερο από το τελικό ΣΣΕ, συνεπώς ο αλγόριθμος υπαγωγής αναμένεται να συμπεριφερθεί καλά στην πράξη. Στη συνέχεια, ο Αλγόριθμος 14 χρησιμοποιεί τις ιδιότητες $(R1)$ και $(R2)$ ως εξής:

- Στη γραμμή 14, *δεν* προσθέτει κανένα ερώτημα $[\bigwedge \mathsf{bd}(\mathcal{Q}') \to \mathcal{Q}_A(nv)]\sigma$ στο $\mathcal{R}'_Q$ εάν το $\mathcal{Q}'$ υπάγεται από κάποιο ερώτημα $\mathcal{Q}''$ στον $\mathcal{G}$ και το $\bigwedge \mathsf{bd}(\mathcal{Q}'') \to \mathcal{Q}_A(nv)$ έχει ήδη προστεθεί στο $\mathcal{R}'_Q$.

- Έστω $\mathcal{Q}_h$ ερώτημα που επιλέγεται στη γραμμή 4. Εάν το $\mathcal{Q}_h$ υπάγεται από κάποιο ερώτημα $\mathcal{Q}'$ στον $\mathcal{G}$ και είτε το $\bigwedge \mathsf{bd}(\mathcal{Q}') \to \mathcal{Q}_A(nv')$ έχει ήδη προστεθεί στο $\mathcal{R}'_Q$, ή ισχύει $\mathcal{Q}' \subseteq \mathcal{Q}_h$, $m(\mathcal{Q}') = m(\mathcal{Q}_h)$, και $\mathsf{canBeJoined}(\mathcal{Q}', m(\mathcal{Q}'), jv) = \mathsf{αληθές}$, τότε ο αλγόριθμος «προσπερνάει» το $\mathcal{Q}_h$—ήτοι, προσθέτει στην Q κάθε ερώτημα $\mathcal{Q}''$ τέτοιο ώστε $\langle \mathcal{Q}_h, \mathcal{Q}'' \rangle \in \mathcal{G}$ και συνεχίζει με το επόμενο ΣΕ.

Αξίζει να σημειωθεί ότι παρόλο που οι συνθήκες της Ιδιότητας $(R2)$ $\mathcal{Q}' \subseteq \mathcal{Q}_h$ και $m(\mathcal{Q}') = m(\mathcal{Q}_h)$ φαίνεται να είναι ιδιαίτερα αυστηρές, στην πραγματικότητα είναι πολύ συχνό το φαινόμενο ένα ερώτημα $\mathcal{Q}'$ να υπάγει ένα ερώτημα $\mathcal{Q}_h$ με βάση την ταυτοτική αντικατάσταση και να ισχύει $m(\mathcal{Q}') = m(\mathcal{Q}) = \emptyset$. Επιπλέον, όσον αφορά στην Ιδιότητα $(R1)$, αξίζει να σημειωθεί ότι στην πλειονότητα των περιπτώσεων το σύνολο μεταβλητών nv που κατασκευάζεται για κάθε ΣΕ στη γραμμή 7 είναι ίδιο για όλα τα ερωτήματα, ενώ στη γραμμή 10 για $\sigma := \{z \mapsto x\}$ συνήθως έχουμε $z \notin \mathsf{var}(\mathcal{Q}')$, συνεπώς στη γραμμή 14 θα έχουμε ότι $[\bigwedge \mathsf{bd}(\mathcal{Q}') \to \mathcal{Q}_A(nv)]\mu' = \bigwedge \mathsf{bd}(\mathcal{Q}') \to \mathcal{Q}_A(nv)$. Παρατηρούμε λοιπόν πως και οι δύο ιδιότητες της Πρότασης 5.2.2 μπορεί να είναι πολύ αποτελεσματικές στην πράξη.

Παράδειγμα 5.2.3. Έστω ότι για κάποιο ΣΕ και TBox, ο Αλγόριθμος 13 έχει υπολογίσει σε κάποιο σημείο το γράφο $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$, όπου το $\mathcal{R}_Q$ περιέχει τα ερωτήματα $\mathcal{Q}_1 = A(x) \wedge R(x,y) \to \mathcal{Q}_A(x)$ και $\mathcal{Q}_2 = A(x) \to \mathcal{Q}_A(x)$ και επίσης $m(\mathcal{Q}_1) = m(\mathcal{Q}_2) = \emptyset$. Είναι προφανές ότι το $\mathcal{Q}_1$ υπάγεται από το $\mathcal{Q}_2$, όμως, ο

αλγόριθμος δεν μπορεί να αφαιρέσει σε αυτό το σημείο το ερώτημα αυτό από τον $\mathcal{G}$. Στο τελικό βήμα, έστω ότι καλείται ο Αλγόριθμος 14 για τον $\mathcal{G}$ και το $\mathcal{R}_\alpha = \{\mathcal{Q}_\alpha\}$, όπου $\mathcal{Q}_\alpha = B(x) \rightarrow \mathcal{Q}_A(x)$. Όταν ο αλγόριθμος προσπελαύνει τα $\mathcal{Q}_1$ και $\mathcal{Q}_\alpha$ είναι εύκολο να δούμε ότι οι συνθήκες της Ιδιότητας $(R2)$ ικανοποιούνται. Πιο συγκεκριμένα, το $\mathcal{Q}_1$ υπάγεται από το $\mathcal{Q}_2$, $\mathcal{Q}_2 \subseteq \mathcal{Q}_1$ και $m(\mathcal{Q}_1) = m(\mathcal{Q}_2)$. Άρα, ο Αλγόριθμος 14 μπορεί να αποφύγει την κατασκευή του ερωτήματος $\mathcal{Q} = A(x) \wedge R(x,y) \wedge B(x) \rightarrow \mathcal{Q}_A(x)$ από τα $\mathcal{Q}_1$ και $\mathcal{Q}_\alpha$. Όντως, σε κάποιο σημείο ο αλγόριθμος θα επιλέξει τα $\mathcal{Q}_2$ και $\mathcal{Q}_\alpha$ και θα παράγει το ερώτημα $\mathcal{Q}' = A(x) \wedge B(x) \rightarrow \mathcal{Q}_A(x)$ που υπάγει το $\mathcal{Q}$, άρα, το ερώτημα $\mathcal{Q}$ όντως πρόκειται να είναι περιττό εάν προστεθεί στο $\mathcal{R}'_Q$. $\diamondsuit$

Εντοπισμός μη-Περιττών Ερωτημάτων

Ιδιαίτερα συχνό είναι το φαινόμενο κατά το οποίο επεκτείνοντας στο i-οστό βήμα ένα ερώτημα που είναι μη-περιττό παράγεται ένα ερώτημα που είναι επίσης μη-περιττό στο βήμα $i+1$. Καθώς συνήθως το μεγαλύτερο μέρος του τελικού ΣΣΕ αποτελείται από ΣΕ που είναι μη-περιττά, θα ήταν ευεργετικό για τον αλγόριθμο μας εάν αυτά μπορούσαν να αναγνωριστούν κατά τη διάρκεια παραγωγής της επανεγγραφής ΣΣΕ του ερωτήματος στον Αλγόριθμου 14. Τα ερωτήματα αυτά μπορούν τότε να μην συμπεριληφθούν στον τελικό έλεγχο απαλοιφής περιττών ερωτημάτων (γραμμή 12 του Αλγορίθμου 13), μειώνοντας έτσι σημαντικά το μέγεθος των βρόχων επανάληψης που χρησιμοποιεί η μέθοδος αυτή για να εκτελεστεί.

Η παρακάτω πρόταση παρέχει μεθόδους για την αναγνώριση μη-περιττών ερωτημάτων κατά τη διάρκεια εκτέλεσης του Αλγόριθμου 14 υποθέτοντας ότι γνωρίζουμε τις σχέσεις υπαγωγής στον γράφο επανεγγραφής της εισόδου.

Πρόταση 5.2.4. *Έστω $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, \mu \rangle$ γράφος επανεγγραφής, έστω $\mathcal{R}_\alpha$ ΣΣΕ, έστω jv, av σύνολα μεταβλητών, και έστω $\mathcal{R}'_Q$ έξοδος του Αλγόριθμου 14 όταν εφαρμόζονται σε αυτόν τα $\mathcal{G}, \mathcal{R}_\alpha, jv$, και av. Έστω ΣΕ $\mathcal{Q} \in \mathcal{R}_Q$ τ.ω.* $\mathsf{canBeJoined}(\mathcal{Q}, m(\mathcal{Q}), jv) =$ αληθές, *έστω nv σύνολο μεταβλητών που κατασκευάζεται για το $\mathcal{Q}$ στη γραμμή 7 και έστω ότι το $\mathcal{Q}$ είναι μη-περιττό στο $\mathcal{R}_Q$. Τότε ισχύουν οι παρακάτω ιδιότητες:*

(N1) *Το $\bigwedge \mathsf{bd}(\mathcal{Q}) \rightarrow \mathcal{Q}_A(nv)$ είναι επίσης μη-περιττό στο $\mathcal{R}'_Q$.*

(N2) *Έστω $\kappa = m(\mathcal{Q})$ και έστω ΣΕ $\mathcal{Q}_\alpha \in \mathcal{R}_\alpha$. Εάν κανένα από τα κατηγορήματα των ατόμων του σώματος του $\mathcal{Q}_\alpha$ δεν εμφανίζεται σε οποιοδήποτε ΣΕ $\mathcal{Q}' \in \mathcal{R}_Q$ και για κάθε $\mathcal{Q}'_\alpha \in \mathcal{R}_\alpha$ έχουμε* $\mathsf{mergeCQs}(\mathcal{Q}'_\alpha, \mathcal{Q}) = \emptyset$, *τότε το ερώτημα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \mathsf{bd}(\mathcal{Q}_\alpha m(\mathcal{Q})) \rightarrow \mathcal{Q}_A(nv)$ είναι επίσης μη-περιττό στο $\mathcal{R}'_Q$.*

Απόδειξη: (Ιδιότητα $(N1)$) Έστω $\mathcal{Q}'$ ένα αυθαίρετο ΣΕ στο $\mathcal{R}_Q$ διαφορετικό από το $\mathcal{Q}$. Κατά τον τερματισμό, είτε ένα ΣΕ της μορφής $[\bigwedge \mathsf{bd}(\mathcal{Q}) \rightarrow \mathcal{Q}_A(nv')]\mu'$ για κάποια μ', είτε κάποιο ΣΕ της μορφής $\bigwedge \mathsf{bd}(\mathcal{Q}') \cup \mathsf{bd}(\mathcal{Q}_\alpha \kappa) \rightarrow \mathcal{Q}_A(nv')$ για κάποιο $\mathcal{Q}_\alpha \in \mathcal{R}_\alpha$ προστίθεται στο $\mathcal{R}'_Q$ από τον Αλγόριθμο 14. Έστω ότι το $[\bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv')]\mu'$ υπάγει το $\bigwedge \mathsf{bd}(\mathcal{Q}) \rightarrow \mathcal{Q}_A(nv)$. Τότε υπάρχει μια αντικατάσταση θ τ.ω. $\mathcal{Q}'\mu' \circ \theta \subseteq \mathcal{Q}$ και συνεπώς, $\mathcal{Q}'\theta' \subseteq \mathcal{Q}$ για $\theta' = \mu' \circ \theta$, αντίθετα με την υπόθεση μας. Το ίδιο ισχύει εάν υποθέσουμε ότι το $\bigwedge \mathsf{bd}(\mathcal{Q}') \cup \mathsf{bd}(\mathcal{Q}_\alpha \kappa) \rightarrow \mathcal{Q}_A(nv')$ υπάγει

το $\mathcal{Q}$. Συνεπώς, μπορούμε να συμπεράνουμε ότι κανένα από τα παραπάνω ερωτήματα δεν μπορεί να υπάγει το $\bigwedge \mathsf{bd}(\mathcal{Q}) \rightarrow \mathcal{Q}_A(nv)$. Εφόσον το $\mathcal{Q}'$ ήταν ένα αυθαίρετο ΣΕ προκύπτει ότι κανένα ερώτημα στο $\mathcal{R}'_Q$ δεν υπάγει το $\bigwedge \mathsf{bd}(\mathcal{Q}) \rightarrow \mathcal{Q}_A(nv)$ και συνεπώς αυτό είναι μη-περιττό στο $\mathcal{R}'_Q$.

(Ιδιότητα $(N2)$) Εφόσον το $\mathcal{Q}$ είναι μη-περιττό στο $\mathcal{R}_Q$ τότε για όλα τα $\mathcal{Q}' \in \mathcal{R}_Q$ διάφορα του $\mathcal{Q}$ και για όλες τις αντικαταστάσεις θ, έχουμε ότι $\mathcal{Q}\theta \not\subseteq \mathcal{Q}$. Έστω ότι το $\mathcal{Q}'$ είναι ένα τέτοιο αυθαίρετο ΣΕ και έστω ότι η θ είναι μια αυθαίρετη αντικατάσταση. Αυτό συνεπάγεται πως πρέπει να υπάρχει κάποιο άτομο At στο σώμα του $\mathcal{Q}'$ τέτοιο ώστε $At\theta \in \mathcal{Q}'\theta$ και $At\theta \notin \mathcal{Q}$. Θεωρούμε τώρα ένα αυθαίρετο ερώτημα $\mathcal{Q}_\alpha \in \mathcal{R}_\alpha$ τέτοιο ώστε κανένα από τα κατηγορήματα των ατόμων του σώματος του να μην εμφανίζονται σε κάποιο άτομο του σώματος του $\mathcal{Q}'$. Αυτό συνεπάγεται ότι $At\theta \notin \mathcal{Q}_\alpha$ και $At\theta \notin \mathcal{Q}_\alpha\kappa$ για κάθε αντικατάσταση κ και συνεπώς επίσης $At\theta \notin \mathcal{Q} \cup \mathcal{Q}_\alpha m(\mathcal{Q})$. Κατά συνέπεια, ούτε το $[\bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv')]\mu'$ για κάθε μ', ούτε το $\bigwedge \mathsf{bd}(\mathcal{Q}') \cup \mathsf{bd}(\mathcal{Q}'_\alpha m(\mathcal{Q}')) \rightarrow \mathcal{Q}_A(nv')$ για κάθε $\mathcal{Q}'_\alpha \in \mathcal{R}_\alpha$ υπάγουν το $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \mathsf{bd}(\mathcal{Q}_\alpha m(\mathcal{Q})) \rightarrow \mathcal{Q}_A(nv)$. Επιπλέον, για κάθε ερώτημα $\mathcal{Q}_\alpha \in \mathcal{R}_\alpha$ έχουμε $\mathsf{mergeCQs}(\mathcal{Q}'_\alpha, \mathcal{Q}) = \emptyset$ και πάλι επειδή κανένα κατηγόρημα από τα άτομα του $\mathcal{Q}'_\alpha$ δεν εμφανίζεται σε κάποιο άλλο ΣΕ, κανένα ερώτημα της μορφής $[\bigwedge \mathsf{bd}(\mathcal{Q}) \rightarrow \mathcal{Q}_A(nv)]\mu'$ για μια αντικατάσταση μ' δεν προστίθεται στο $\mathcal{R}'_Q$ από τον Αλγόριθμο 14. Συνοψίζοντας, το $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \mathsf{bd}(\mathcal{Q}_\alpha\mu(\mathcal{Q})) \rightarrow \mathcal{Q}_A(nv)$ είναι μη-περιττό στο $\mathcal{R}'_Q$. □

Έστω $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$ γράφος και $\mathcal{R}_\alpha$ ΣΣΕ με τα οποία καλείται ο Αλγόριθμος 14. Τότε, για να μπορούν να χρησιμοποιηθούν οι Ιδιότητες $(N1)$ και $(N2)$ ο αλγόριθμος τροποποιείται ως εξής:

- Στην αρχή αρχικοποιεί ένα άδειο σύνολο NR που περιέχει μη-περιττά ερωτήματα.
- Στη γραμμή 14, εάν $[\bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv)]\mu' = \bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv)$, και επίσης το $\mathcal{Q}'$ είναι μη-περιττό στο $\mathcal{R}_Q$ προσθέτει το $[\bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv)]\mu'$ στο NR.
- Στη γραμμή 8, προσθέτει το $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \mathsf{bd}(\mathcal{Q}_\alpha m(\mathcal{Q})) \rightarrow \mathcal{Q}_A(nv)$ στο NR εάν το $\mathcal{Q}_h$ είναι μη-περιττό στο $\mathcal{R}_Q$, κανένα από τα κατηγορήματα στο $\mathcal{Q}_\alpha$ δεν εμφανίζεται σε κάποιο ερώτημα στο $\mathcal{R}_Q$ και εάν για κάθε $\mathcal{Q}'_\alpha \in \mathcal{R}_\alpha$ έχουμε $\mathsf{mergeCQs}(\mathcal{Q}'_\alpha, \mathcal{Q}) = \emptyset$.
- Τέλος, επιστρέφει το ΣΣΕ $\mathcal{R}'_Q$ και το σύνολο NR.

Στη συνέχεια, το σύνολο NR που έχει επιστραφεί χρησιμοποιείται από τη μέθοδο removeRedundant στη γραμμή 12 του Αλγόριθμου 13 ως εξής: Όλα τα ερωτήματα που ανήκουν στο σύνολο NR αποκλείονται από τον έλεγχο απαλοιφής όπως φαίνεται και από τον Αλγόριθμο 7.

Αξίζει να σημειωθεί και πάλι ότι ο έλεγχος για το εάν για κάθε $\mathcal{Q}_\alpha$ δεν υπάρχει κανένα κατηγόρημα σε κάποιο άτομο του σώματος ερωτήματος στο $\mathcal{R}_Q$ απαιτεί την επανάληψη για κάθε $\mathcal{Q}_\alpha \in \mathcal{R}_\alpha$ από όλα τα ερωτήματα στο $\mathcal{R}_Q$. Παρόλα αυτά, αυτό μπορεί να πραγματοποιηθεί μόνο μια φορά στην αρχή του Αλγόριθμου 14 και επιπλέον, όπως έχει αναφερθεί και προηγουμένως, στο σημείο αυτό του αλγόριθμου το σύνολο $\mathcal{R}_Q$ αναμένεται να μην είναι πολύ μεγάλο σε μέγεθος. Επιπλέον, όπως θα δείξουμε και

στο Κεφάλαιο 6 σε πολλές περιπτώσεις η τεχνική αυτή μας επιτρέπει να μειώσουμε αισθητά το κόστος ελέγχου για περιττά ερωτήματα στο σύνολο $\mathcal{R}'_Q$ και σε πολλές περιπτώσεις να το αποφύγουμε εντελώς, εάν $\mathcal{R}'_Q = NR$. Το όφελος αυτό υπερτερεί σημαντικά του χρόνου που απαιτείται για όλους τους επιπλέον ελέγχους που εισάγει η τεχνική αυτή.

Παράδειγμα 5.2.5. Έστω και πάλι τα $\mathcal{G}, \mathcal{R}_Q$, και $\mathcal{R}_\alpha$ από το Παράδειγμα 5.2.3. Εάν το $\mathcal{R}_Q$ δεν περιέχει άλλα ερωτήματα, τότε είναι εύκολο να δούμε ότι για το $\mathcal{Q}_2 \in \mathcal{R}_Q$ και το $\mathcal{Q}_\alpha \in \mathcal{R}_\alpha$ οι συνθήκες της Ιδιότητας $(N2)$ ικανοποιούνται. Πιο συγκεκριμένα, το $\mathcal{Q}_2$ είναι μη-περιττό στο $\mathcal{R}_Q$, κανένα από τα κατηγορήματα του $\mathcal{Q}_\alpha$ δεν εμφανίζονται σε άλλα ΣΕ στο $\mathcal{R}_Q$ και $\mathsf{mergeCQs}(\mathcal{Q}_\alpha, \mathcal{Q}_2) = \emptyset$. Συνεπώς, το ΣΕ που παράγεται από το $\mathcal{Q}_2$ και το $\mathcal{Q}_\alpha$—δηλαδή, το $\mathcal{Q}'$ από το Παράδειγμα 5.2.3, είναι μη-περιττό στο αποτέλεσμα. Παρόλα αυτά, έστω ότι εκτός από τα $\mathcal{Q}_1$ και $\mathcal{Q}_2$ η επανεγγραφή ΣΣΕ $\mathcal{R}_Q$ περιέχει επίσης το ΣΕ $\mathcal{Q}_3 = S(x,z) \rightarrow \mathcal{Q}_A(x)$ και το $\mathcal{R}_\alpha$ περιέχει επίσης το ΣΕ $\mathcal{Q}'_\alpha = S(x,z) \rightarrow \mathcal{Q}_A(x)$. Η Ιδιότητα $(N2)$ δεν ικανοποιείται τώρα για τα $\mathcal{Q}_2$ και $\mathcal{Q}'_\alpha$, εφόσον το κατηγόρημα του $\mathcal{Q}'_\alpha$ εμφανίζεται στο $\mathcal{Q}_3$. Άρα, δεν μπορούμε να εγγυηθούμε ότι το $\mathcal{Q}_2$ και το $\mathcal{Q}'_\alpha$ θα παράγουν ένα μη-περιττό ερώτημα. Στην πραγματικότητα, το ερώτημα που κατασκευάζουν (ήτοι, το $A(x) \wedge S(x,z) \rightarrow \mathcal{Q}_A(x)$) πρόκειται να υπαχθεί από το ερώτημα που παράγεται από το $\mathcal{Q}_3$ και το $\mathcal{Q}'_\alpha$ (ήτοι, το $S(x,z) \rightarrow \mathcal{Q}_A(x)$). Το ερώτημα $\mathcal{Q}'$ όμως που παράγεται από τα $\mathcal{Q}_2$ και $\mathcal{Q}_\alpha$ παραρμένει μη-περιττό. $\diamondsuit$

Εντοπισμός μη-Υπαγόντων Ερωτημάτων

Τέλος η επόμενη πρόταση παρουσιάζει μια ιδιότητα με χρήση της οποίας μπορούμε να αναγνωρίσουμε εάν ένα ερώτημα που παράγεται κατά την εκτέλεση του Αλγορίθμου 14 δεν θα υπάγει κανένα άλλο ΣΕ που παράγεται από τον ίδιο αλγόριθμο.

Πρόταση 5.2.6. *Έστω $\mathcal{G} = \langle \mathcal{R}_Q, \mathcal{H}, m \rangle$ γράφος επανεγγραφής, έστω $\mathcal{R}_\alpha$ ΣΣΕ, έστω jv, av σύνολα μεταβλητών, και έστω $\mathcal{R}'_Q$ έξοδος του Αλγορίθμου 14 όταν εφαρμόζεται στα $\mathcal{G}, \mathcal{R}_\alpha, jv$, και av. Έστω ΣΕ $\mathcal{Q} \in \mathcal{R}_Q$ με* $\mathsf{canBeJoined}(\mathcal{Q}, m(\mathcal{Q}), jv) =$ αληθές, *έστω nv σύνολο μεταβλητών που κατασκευάζεται για το $\mathcal{Q}$ στη γραμμή 7, και έστω ότι το $\mathcal{Q}$ δεν υπάγει κανένα ΣΕ στο $\mathcal{R}_Q$. Τότε ισχύει η παρακάτω ιδιότητα:*

(S) *Έστω ΣΕ $\mathcal{Q}_\alpha \in \mathcal{R}_\alpha$. Εάν κανένα από τα κατηγορήματα των ατομων του σώματος του $\mathcal{Q}_\alpha$ δεν εμφανίζεται σε κάποιο ΣΕ $\mathcal{R}_Q$ (συμπεριλαμβανομένου του $\mathcal{Q}$), τότε το ερώτημα $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \mathsf{bd}(\mathcal{Q}_\alpha m(\mathcal{Q})) \rightarrow \mathcal{Q}_A(nv)$ δεν υπάγει κανένα ΣΕ στο $\mathcal{R}'_Q$.*

Απόδειξη: Εφόσον το $\mathcal{Q}$ δεν υπάγει κανένα ΣΕ στο $\mathcal{R}_Q$, έχουμε $\mathcal{Q}\theta \not\subseteq \mathcal{Q}'$ για όλα τα $\mathcal{Q}' \in \mathcal{R}_Q$ που είναι διαφορετικά από το $\mathcal{Q}$ και τις αντικαταστάσεις θ. Συνεπώς, για κάθε θ υπάρχει ένα άτομο At στο σώμα του $\mathcal{Q}$ τ.ω. $At\theta \in \mathcal{Q}\theta$ και $At\theta \notin \mathcal{Q}'$. Για να δείξουμε ότι το $\bigwedge \mathsf{bd}(\mathcal{Q}) \cup \mathsf{bd}(\mathcal{Q}_\alpha m(\mathcal{Q})) \rightarrow \mathcal{Q}_A(nv)$, που θα καλείται $\mathcal{Q}_n$ στη συνέχεια, δεν υπάγει κανένα ΣΕ στο $\mathcal{R}'_Q$ θεωρούμε ένα τυχαίο ΣΕ $\mathcal{Q}' \in \mathcal{R}_Q$. Κατά τον τερματισμό, ο Αλγόριθμος 14 μπορεί να προσθέσει στο $\mathcal{R}'_Q$ είτε το ΣΕ $[\bigwedge \mathsf{bd}(\mathcal{Q}') \rightarrow \mathcal{Q}_A(nv')]\mu'$ για κάποια μ' ή το ΣΕ $\bigwedge \mathsf{bd}(\mathcal{Q}') \cup \mathsf{bd}(\mathcal{Q}'_\alpha \kappa) \rightarrow \mathcal{Q}_A(nv')$ για κάποιο (πιθανώς διαφορετικό) $\mathcal{Q}'_\alpha \in \mathcal{R}_\alpha$. Θα δείξουμε ότι το $\mathcal{Q}_n$ δεν υπάγει κανένα τέτοιο ΣΕ.

Από την υπόθεση κανένα από τα κατηγορήματα των ατόμων του $\mathcal{Q}_\alpha$ δεν εμφανίζεται στο $\mathcal{Q}'$. Άρα, είναι ξεκάθαρο ότι υπάρχει κάποιο $At' \in \mathcal{Q}_n$ (κάποιο που εμφανίζεται στο $\mathcal{Q}_\alpha$) τέτοιο ώστε για κάθε θ να έχουμε $At'\theta \notin [\bigwedge \mathsf{bd}(\mathcal{Q}') \to \mathcal{Q}_A(nv')]\mu'$, συνεπώς, το $\mathcal{Q}_n$ δεν υπάγει το $[\bigwedge \mathsf{bd}(\mathcal{Q}') \to \mathcal{Q}_A(nv')]\mu'$. Επιπλέον, από τα παραπάνω προκύπτει ότι το $\mathcal{Q}_n$ δεν υπάγει το $\bigwedge \mathsf{bd}(\mathcal{Q}') \cup \mathsf{bd}(\mathcal{Q}'_\alpha\kappa) \to \mathcal{Q}_A(nv')$ για κάθε κ και κάθε $\mathcal{Q}'_\alpha \in \mathcal{R}_\alpha$ διάφορο του $\mathcal{Q}_\alpha$.

Τέλος, έστω το ΣΕ $\bigwedge \mathsf{bd}(\mathcal{Q}') \cup \mathsf{bd}(\mathcal{Q}_\alpha\kappa) \to \mathcal{Q}_A(nv')$. Θυμίζουμε ότι για κάθε θ υπάρχει $At\theta \in \mathcal{Q}\theta$ τέτοιο ώστε $At\theta \notin \mathcal{Q}'$. Συνεπώς, πάλι λόγω της υπόθεσης εφόσον κανένα από τα άτομα του $\mathcal{Q}_\alpha$ δεν εμφανίζεται στο $\mathcal{Q}$ έχουμε ότι $At\theta \notin \mathcal{Q}_\alpha$ και άρα ισχύει επίσης $At\theta \notin \bigwedge \mathsf{bd}(\mathcal{Q}') \cup \mathsf{bd}(\mathcal{Q}_\alpha\kappa) \to \mathcal{Q}_A(nv')$.

Εφόσον το $\mathcal{Q}'$ ήταν ένα τυχαίο ερώτημα, προκύπτει ότι το $\mathcal{Q}_n$ δεν μπορεί να υπάγει κανένα από τα ΣΕ που παράγονται από τον Αλγόριθμο 14. □

Όμοια με πριν, ο Αλγόριθμος 14 χρησιμοποιεί επιπλέον σύνολα για να αποθηκεύσει τέτοια ερωτήματα που στη συνέχεια εξαιρούνται από τον βρόχο της συνάρτησης removeRedundant όπως φαίνεται στον Αλγόριθμο 7.

Παράδειγμα 5.2.7. Θεωρούμε και πάλι τα Παραδείγματα 5.2.3 και 5.2.5, και την επανεγγραφή ΣΣΕ $\mathcal{R}_Q = \{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3\}$. Είναι εύκολο να δούμε ότι για το $\mathcal{Q}_1 \in \mathcal{R}_Q$ και για το $\mathcal{Q}_\alpha \in \mathcal{R}_\alpha$ οι συνθήκες της Ιδιότητας (S) ικανοποιούνται. Πιο συγκεκριμένα, το $\mathcal{Q}_1$ δεν υπάγει κανένα ΣΕ στην $\mathcal{R}_Q$ και κανένα από τα κατηγορήματα του $\mathcal{Q}_\alpha$ δεν εμφανίζεται σε κανένα ΣΕ στην $\mathcal{R}_Q$. Συνεπώς, το ΣΕ που παράγεται από το $\mathcal{Q}_1$ και το $\mathcal{Q}_\alpha$—που είναι το $\mathcal{Q}' = A(x) \wedge R(x,y) \wedge B(x) \to \mathcal{Q}_A(x)$ δεν πρόκειται να υπάγει κανένα ΣΕ από το αποτέλεσμα. Παρόλα αυτά, έστω ότι το σύνολο $\mathcal{R}_\alpha$ περιέχει επίσης το ΣΕ $\mathcal{Q}'_\alpha = R(x,z) \to \mathcal{Q}_A(x)$. Η Ιδιότητα (S) δεν ικανοποιείται τώρα για τα $\mathcal{Q}_1$ και $\mathcal{Q}'_\alpha$, εφόσον το κατηγόρημα του $\mathcal{Q}'_\alpha$ εμφανίζεται στο $\mathcal{Q}_1$. Άρα, δεν μπορούμε να εγγυηθούμε ότι τα $\mathcal{Q}_1$ και $\mathcal{Q}'_\alpha$ θα παράγουν ένα ερώτημα που δεν πρόκειται να υπάγει κανένα άλλο ερώτημα από το αποτέλεσμα. Στην πραγματικότητα το ερώτημα που παράγουν, δηλαδή, το $A(x) \wedge R(x,y) \wedge R(x,z) \to \mathcal{Q}_A(x)$ υπάγει το ερώτημα που παράγειται από το $\mathcal{Q}_2$ και το $\mathcal{Q}'_\alpha$ δηλαδή, το $A(x) \wedge R(x,z) \to \mathcal{Q}_A(x)$. ◇

Μέρος III

Πρακτικά Ζητήματα

Κεφάλαιο 6

Αξιολόγηση

Στο κεφάλαιο αυτό παρουσιάζουμε την αξιολόγηση των αλγορίθμων που περιγράψαμε στα προηγούμενα κεφάλαια έτσι ώστε να μπορέσουμε να οδηγηθούμε σε χρήσιμα συμπεράσματα όσον αφορά στην αποδοτικότητα και χρησιμότητά τους. Αρχικά στην ενότητα 6.1 περιγράφουμε το σύνολο δεδομένων που χρησιμοποιήσαμε για την αξιολόγηση μας. Στην ενότητα 6.2 παρουσιάζουμε τα πειραματικά μας αποτελέσματα για τους Αλγορίθμους 3–5 που αφορούν στην κατασκευή επανεγγραφής για ερωτήματα τα οποία έχουν τροποποιηθεί με την προσθήκη/αφαιρεση διακεκριμένων μεταβλητών ή με την αφαίρεση ατόμων για οντολογίες την εκφραστικότητα των οποίων καταγράφουν οι υπαρξιακοί κανόνες. Στη συνέχεια, στην ενότητα 6.3 παρουσιάζουμε τα αποτελέσματα της αξιολόγησης των Αλγορίθμων 9–14 που αφορούν στην κατασκευή ενός συστήματος επαυξητικής επανεγγραφής για ένα σταθερό ερώτημα για DL-Lite$_R$ οντολογίες. Τα αποτελέσματα και από τις δύο αξιολογήσεις είναι ιδιαίτερα ενθαρρυντικά.

Όλα τα πειράματα πραγματοποιήθηκαν σε έναν υπολογιστή MacBook Pro με επεξεργαστή στα 2.66GHz και με 4GB μνήμης RAM, με ένα χρονικό όριο 600 δευτερολέπτων.

6.1 Σύνολο Δεδομένων Αξιολόγησης

Για την αξιολόγηση των αλγορίθμων μας χρησιμοποιήσαμε δύο πλαίσια ανάλογα με την εκφραστικότητα των οντολογιών που περιέχουν. Το πρώτο αφορά σε οντολογίες εκφραστικότητας DL-Lite, ενώ το δεύτερο σε οντολογίες εκφραστικότητας $\mathcal{ELHI}$.

Το πρώτο πλαίσιο που αφορά σε οντολογίες DL-Lite είναι το πλαίσιο που προτάθηκε από τους Hector et al. [116] και αποτελείται από εννέα οντολογίες και πέντε ερωτήματα για κάθε οντολογία. Η οντολογία V καταγράφει πληροφορίες σχετικά με την Ευρωπαϊκή ιστορία και έχει αναπτυχθεί στα πλαίσια του έργου VICODI,[1]. Οι P1 και P5 αποτελούν δύο τεχνητές οντολογίες οι οποίες παρέχουν ένα ελεγχόμενο σενάριο για την μελέτη του βήματος μείωσης του αλγορίθμου PerfectRef. Η S μοντελοποιεί πληροφορίες σχετικά με οικονομικά ιδρύματα της Ευρωπαϊκής Ένωσης και έχει

[1] http://www.vicodi.org/

Πίνακας 6.1: Αριθμός εννοιών, ρόλων και αξιωμάτων των δεδομένων ελέγχου

	DL-Lite									$\mathcal{ELHI}$			
	V	P1	P5	P5X	S	U	UX	A	AX	L	G_1	G_2	G_3
Έννοιες	194	2	6	6	18	34	35	74	74	43	99	59	83
Ρόλοι	10	1	1	5	12	26	31	5	31	32	46	29	65
Αξιώματα	222	2	10	18	51	127	137	137	189	93	169	119	221

αναπτυχθεί για την μελέτη της πρόσβασης σε δεδομένα πάνω από οντολογίες [128]. Η U είναι μια DL-Lite$_R$ έκδοση της γνωστής οντολογίας LUBM[2] που κατασκευάστηκε από το Πανεπιστήμιο του Lehign και περιγράφει την δομή οργάνωσης των πανεπιστημίων. Η A είναι μια οντολογία που καταγράφει πληροφορίες σχετικά με ικανότητες και αναπηρίες. Επιπρόσθετα, χρησιμοποιήσαμε ακόμα τις οντολογίες P5X, UX και AX που αποτελούν κανονικοποιημένες εκδόσεις των οντολογιών P5, U και A αντίστοιχα, όσον αφορά στην ύπαρξη του προσδιορισένου υπαρξιακού ποσοδείκτη στο δεξί μέρος των αξιωμάτων υπαγωγής. Αξίζει να σημειωθεί ότι τα αποτελέσματα για την οντολογία P1 κατά τη διάρκεια της αξιολόγησης μας συχνά παραλείπονται μιας και είναι τετριμμένα.

Το δεύτερο πλαίσιο που αφορά σε οντολογίες $\mathcal{ELHI}$ περιλαμβάνει τη συγκριτική αξιολόγηση της οντολογίας LUBM [68], που αποτελείται από μια οντολογία (L) και 14 ερωτήματα καθώς και τρία τμήματα, εκφραστικότητας $\mathcal{ELHI}$, από την οντολογία GALEN (G), που είναι μια ευρέως γνωστή βιοϊατρική οντολογία με σύνθετη δομή, και ονομάζονται G_1, G_2, και G_3.[3] Για τις G_1, G_2, και G_3 κατασκευάσαμε από πέντε ερωτήματα.

Στον Πίνακα 6.1 παραθέτουμε τον αριθμό των εννοιών, ρόλων και αξιωμάτων για τις οντολογίες που απαρτίζουν το σύνολο δεδομένων μας. Τα ερωτήματα και από τα δύο πλαίσια παρατίθενται στο Παράρτημα Α΄.

6.2 Αξιολόγηση Επανεγγραφής Τροποποιημένων Ερωτημάτων

Για την πειραματική αυτή αξιολόγηση υλοποιήσαμε τους Αλγορίθμους 3–5 σε πρωτότυπα συστήματα. Για να υλοποιήσουμε τη συνάρτηση rew που χρησιμοποιείται, πραγματοποιήσαμε μια υλοποίηση χρησιμοποιώντας το σύστημα Rapid [40], που αποτελεί ένα πολύ γρήγορο σύστημα επανεγγραφής για οντολογίες DL-Lite, και μια υλοποίηση χρησιμοποιώντας το Requiem, ένα σύστημα επανεγγραφής για την πιο εκφραστική γλώσσα $\mathcal{ELHI}$ [116]. Στη συνέχεια θα καλούμε την πρώτη υλοποίηση Ref$_L$ και την δεύτερη Ref$_E$. Τα αποτελέσματα της υλοποίησης Ref$_L$ τα συγκρίναμε με το Rapid και της Ref$_E$ με το Requiem.

Όλοι οι πίνακες που παρουσιάζουμε σε αυτή την ενότητα χρησιμοποιούν την ίδια σημειολογία. Με $\sharp\mathcal{R}$ δηλώνουμε το μέγεθος της επανεγγραφής που χρησιμοποιείται

[2] http://swat.cse.lehigh.edu/projects/lubm/

[3] Τα τμήματα αυτά είναι διαθέσιμα στον http://image.ece.ntua.gr/~avenet/CQ_Refinement

Πίνακας 6.2: Αξιολόγηση του Αλγορίθμου 3 χρησιμοποιώντας DL-Lite οντολογίες

$\mathcal{O}$	Q	Ref_L, Αλγόριθμος 3				Rapid	
		$\sharp\mathcal{R}$	t_{rew}	t_{alg}	t_{sub}	t_{R}	t_{sub}
V	2	10	5,0	7,0	2,0	17,5	0,5
	3	72	0,0	2,0	8,0	14,5	0,0
	4	185	0,0	4,5	27,5	25,5	0,5
S	2	7	1,5	2,0	0,5	19,0	1,0
	3	8	13,8	14,3	0,8	6,3	0,2
	4	11	8,3	8,7	0,2	19,2	0,3
	5	14	16,7	17,0	0,9	39,9	1,1
U	2	3	0,5	0,5	0,0	3,0	0,0
	3	8	3,3	3,5	0,0	4,2	0,0
	4	8	11,0	11,0	0,0	12,5	0,0
UX	2	3	0,0	0,0	0,0	4,5	0,0
	3	16	0,0	0,2	0,3	10,5	0,0
	4	11	0,5	0,5	0,5	26,0	0,0

ως είσοδος στους αλγορίθμους μας, με t_{alg} δηλώνουμε τον χρόνο που χρειάζονται οι Αλγόριθμοι 3–5 για να ολοκληρωθούν, χωρίς όμως να λαμβάνουμε υπ' όψιν το τελικό βήμα απαλοιφής περιττών ερωτημάτων. Επιπρόσθετα, για τους πίνακες που αναφέρονται στους Αλγόριθμους 3 και 5, δηλώνουμε με t_{rew} τον χρόνο που χρειάζεται για να τρέξει η υπο-ρουτίνα rew. Όσον αφορά τους αλγόριθμος Rapid και Requiem, με t_{R} δηλώνουμε τον χρόνο που χρειάζονται για να ολοκληρωθούν χωρίς όμως να υπολογίζουμε την διάρκεια του βήματος απαλοιφής περιττών ερωτημάτων. Για όλα τα συστήματα τέλος, δηλώνουμε με t_{sub} τον χρόνο που χρειάζεται η μέθοδος απαλοιφής περιττών ερωτημάτων. Μετά και την εφαρμογή αυτού του βήματος όλα τα συστήματα επιστρέφουν επανεγγραφές που έχουν ίδιο μέγεθος και για το λόγο αυτό τα αποτελέσματα αυτά δεν παρουσιάζονται. Όλοι οι χρόνοι είναι μετρημένοι σε χιλιοστά του δευτερολέπτου.

6.2.1 Μείωση Ερωτημάτων με Διακεκριμένες Μεταβλητές

Για την αξιολόγηση του Αλγορίθμου 3 χρησιμοποιήσαμε μόνο εκείνα τα ερωτήματα που έχουν περισσότερες από μια διακεκριμένες μεταβλητές. Για κάθε ένα από αυτά εκτελέσαμε αρκετές επαναλήψεις αφαιρώντας κάθε φορά μια διαφορετική μεταβλητή, συνεπώς οι χρόνοι που παρουσιάζονται δείχνουν μέσους χρόνους.

Ο Πίνακας 6.2 παρουσιάζει τη σύγκριση μεταξύ της Ref_L και του Rapid για το πλαίσιο που αφορά στις DL-Lite οντολογίες. Όπως μπορούμε να παρατηρήσουμε η Ref_L είναι γενικά πιο γρήγορη από το Rapid, με εξαίρεση το ερώτημα 4 της οντολογίας V στο οποίο ο χρόνος t_{sub} για την Ref_L είναι μεγαλύτερος από το συνολικό χρόνο που χρειάζεται το Rapid. Αξιοσημείωτες περιπτώσεις είναι το ερώτημα 5 της οντολογίας S και τα ερωτήματα 3 και 4 της οντολογίας UX. Ο λόγος είναι ότι ο Αλγόριθμος 3 προσπελαύνει τα ερωτήματα από μια σχετικά μικρή επανεγγραφή εφαρμόζοντας σε αυτά τον τελεστή προβολής. Επιπλέον, παρατηρούμε πως η υπο-ρουτίνα rew χρειάζεται πάρα πολύ λίγο χρόνο για την εκτέλεση της. Ενδιαφέρον είναι επίσης ότι συνήθως ο

Πίνακας 6.3: Αξιολόγηση του Αλγορίθμου 3 χρησιμοποιώντας $\mathcal{ELHI}$ οντολογίες

$\mathcal{O}$	Q	Ref_E, Αλγόριθμος 3				Requiem	
		$\sharp\mathcal{R}$	t_{rew}	t_{alg}	t_{sub}	t_{R}	t_{sub}
L	1	3	42,5	42,5	0,0	159,0	110,0
	2	10	33,0	33,7	2,0	88,2	3,5
	3	1	0,0	0,0	0,0	1,5	1,0
	4	348	38,9	39,8	1,7	27,6	29,8
	5	334	31,0	33,0	2,5	46,0	56,0
	7	334	110,5	110,5	4,3	15,5	10,3
	8	347	19,7	20,3	0,6	15,9	11,4
	9	-	-	-	-	-	-
	10	331	10,5	10,5	0,5	16,5	10,0
	11	3	0,0	0,0	0,0	0,0	0,0
	12	333	13,3	13,3	0,2	14,8	11,5
	13	330	0,5	0,5	0,0	27,5	20,5
G_1	1	10	101,0	101,5	4,5	658,0	100,0
	2	62	185,0	186,0	21,0	266,0	106,5
	3	124	180,0	181,5	18,0	361,0	134,5
	4	4	84,5	84,5	2,0	126,5	30,5
	5	12	76,0	76,0	1,5	121,5	0,5
G_2	1	373	1 591,0	1 591,5	1,0	4 038,5	68,0
	2	395	2 818,0	2 818,5	1,5	3 835,5	24,0
	3	484	1 574,0	1 575,5	5,0	3 850,0	28,0
	4	685	1 688,0	1 692,5	43,0	4 149,0	63,0
	5	443	1 632,5	1 634,0	1,5	3 889,5	16,0
G_3	1	11	72,5	72,5	1,0	334,5	24,0
	2	128	112,0	114,8	4,7	163,3	18,7
	3	63	20,0	20,5	1,5	101,5	3,5
	4	79	96,2	96,5	1,4	82,2	3,3
	5	165	19,0	19,0	0,5	47,5	5,0

χρόνος t_{rew} καταλαμβάνει το μεγαλύτερο μέρος του χρόνου εκτέλεσης του αλγορίθμου μας.

Ο Πίνακας 6.3 παρουσιάζει τη σύγκριση μεταξύ της Ref_E και του Requiem για το πλαίσιο που αφορά στις $\mathcal{ELHI}$ οντολογίες. Για το ερώτημα 9 το Requiem δεν μπορούσε να παράγει επανεγγραφή μέσα στο χρονικό όριο των 600 δευτερολέπτων και συνεπώς δεν μπορούσαμε να τρέξουμε την Ref_E, αφού δεν υπήρχε διαθέσιμη επανεγγραφή εισόδου. Και πάλι παρατηρούμε ότι το σύστημα μας είναι αισθητά γρηγορότερο[4] από το Requiem, με πιο επιλήψιμες περιπτώσεις τα ερωτήματα 1, 2, 5 και 13 για την οντολογία LUBM με το τελευταίο να είναι είναι περίπου 25 φορές γρηγορότερο, καθώς και για τα ερωτήματα 1 και 3 της οντολογίας G_1, όλα τα ερωτήματα της οντολογίας G_2 και τα ερωτήματα 1, 3 και 5 της οντολογίας G_3. Όπως και παραπάνω ο λόγος για αυτό είναι ότι ο αλγόριθμος μας προσπελαύνει ερωτήματα από μια σχετικά μικρή επανεγγραφή και πραγματοποιεί απλές λειτουργίες. Παρόλα αυτά, υπάρχουν μερικές περιπτώσεις όπου το Requiem είναι αισθητά γρηγορότερο, όπως τα ερωτήματα 4, 7, και 8 της οντολογίας L και το ερώτημα 4 της οντολογίας G_3. Με προσεκτική ανάλυση συμπεράναμε ότι στα πειράματα αυτά ο αλγόριθμος μας εκτελούσε την υπο-

[4] Ένα σύστημα X θεωρείται «αισθητά γρηγορότερο» από ένα σύστημα Y εάν $t_Y - t_X > 20\text{ms}$

Πίνακας 6.4: Αξιολόγηση του Αλγορίθμου 4 χρησιμοποιώντας DL-Lite οντολογίες.

$\mathcal{O}$	Q	Ref_L, Αλγόριθμος 4			Rapid	
		$\sharp\mathcal{R}$	t_{alg}	t_{sub}	t_{R}	t_{sub}
V	2	10	1,0	0,0	56,0	1,0
	5	30	3,1	1,6	27,4	0,0
P5	1	6	0,0	0,0	0,0	0,0
	2	10	0,7	0,0	1,7	0,0
	3	13	0,3	0,1	3,3	0,1
	4	15	0,3	0,0	11,5	0,3
	5	16	0,1	0,1	22,0	0,6
P5X	1	14	0,0	0,0	1,0	0,0
	2	81	0,3	0,0	11,0	0,0
	3	413	1,6	1,0	15,9	2,3
	4	2 070	10,1	3,7	62,9	64,5
	5	10 352	70,0	19,0	126,1	94,9
U	1	2	0,0	0,0	0,0	0,0
	5	3 375	22,0	10,0	2,0	0,0
UX	1	5	0,0	0,0	0,3	0,0
	5	8 955	27,0	26,0	2,0	0,0
A	1	378	1,0	0,0	2,0	0,0
	2	103	1,0	0,0	1,0	0,0
	3	104	1,0	1,0	3,0	0,3
	4	471	3,0	2,0	16,0	1,0
	5	624	5,7	2,3	20,0	1,7
AX	1	794	0,0	0,0	5,0	1,0
	2	1 812	3,0	8,0	40,0	3,0
	3	4 763	19,0	44,0	122,0	2,3
	4	7 229	19,0	26,0	77,0	2,0
	5	78 885	661,3	777,3	742,7	31,7

ρουτίνα rew αρκετές φορές. Για παράδειγμα, στα ερωτήματα 7 και 8 της οντολογίας L η υπο-ρουτίνα rew καλείται κατά μέσο όρο 4 φορές σε κάθε περίπτωση, σε αντίθεση με ένα μέσο όρο μικρότερο της μιας φοράς για τα υπόλοιπα ερωτήματα της. Αξίζει να σημειωθεί τέλος ότι ο χρόνος t_{rew} σε μερικές περιπτώσεις είναι μεγαλύτερος από το χρόνο εκτέλεσης του Requiem.

6.2.2 Επέκταση Ερωτημάτων με Διακεκριμένες Μεταβλητές

Για την αξιολόγηση αυτή χρησιμοποιήσαμε όλα τα προτεινόμενα ερωτήματα που έχουν περισσότερες από δύο μεταβλητές. Εκτελέσαμε αρκετές επαναλήψεις επεκτείνοντας κάθε φορά κάθε ερώτημα με μια διαφορετική μεταβλητή. Συνεπώς, και πάλι οι χρόνοι που αναφέρονται στους πίνακες είναι μέσοι όροι.

Ο Πίνακας 6.4 παρουσιάζει την σύγκριση ανάμεσα στην Ref_L και το Rapid για το πλαίσιο που αφορά στις DL-Lite οντολογίες. Παρατηρούμε ότι ο αλγόριθμος μας είναι αισθητά γρηγορότερος στην πλειοψηφία των ερωτημάτων κάτι το οποίο μπορεί να αιτιολογηθεί από την πολύ απλή δομή του Αλγορίθμου 4 και το γεγονός ότι ο αλγόριθμος αυτός δεν χρειάζεται να καλέσει την υπο-ρουτίνα rew. Επιπλέον, παρα-

Πίνακας 6.5: Αξιολόγηση του Αλγορίθμου 4 χρησιμοποιώντας $\mathcal{ELHI}$ οντολογίες

$\mathcal{O}$	Q	Ref_E, Αλγόριθμος 4			Requiem	
		$\sharp\mathcal{R}$	t_{alg}	t_{sub}	t_{R}	t_{sub}
L	1	5	1,0	0,0	192,0	1,0
	2	32	0,7	1,3	105,0	20,0
	3	17	0,0	0,0	1,0	2,0
	4	155	0,0	1,0	33,0	196,0
	5	158	1,0	0,0	24,0	17,0
	7	27	0,0	0,0	14,0	11,0
	8	23	0,0	0,0	18,0	12,0
	9	-	-	-	-	-
	10	11	0,0	0,0	15,0	13,0
	11	3	0,0	0,0	0,0	0,0
	12	8	0,0	0,0	24,0	11,0
	13	191	1,0	1,0	27,0	20,0
G_1	1	22	2,0	0,0	965,0	2,0
	2	439	2,0	0,0	360,0	10,0
	3	783	3,0	1,0	221,0	15,0
	4	7	0,0	1,0	144,0	1,0
	5	21	1,0	0,0	134,0	1,0
G_2	1	98	1,0	0,0	3 488,0	78,0
	2	60	0,0	0,0	2 802,0	51,0
	3	461	2,0	1,0	2 906,0	33,0
	4	622	3,0	3,0	3 151,0	64,0
	5	210	0,0	1,0	2 962,0	22,0
G_3	1	14	1,0	0,0	389,0	2,0
	2	323	3,0	2,0	279,5	67,0
	3	91	1,0	1,0	130,0	3,0
	4	139	1,0	0,2	105,3	4,3
	5	60	0,0	1,0	57,0	8,0

τηρούμε ότι οι τεχνικές που χρησιμοποιούνται για την βελτιστοποίηση του βήματος απαλοιφής περιττών ερωτημάτων, και παρουσιάζονται στην ενότητα 3.5 βελτιώνουν κατά πολύ την επίδοση της συνάρτησης απαλοιφής περιττών ερωτημάτων. Το γεγονός αυτό φαίνεται ιδιαίτερα στα ερωτήματα 4 και 5 της οντολογίας P5X, όπου το βήμα απαλοιφής περιττών ερωτημάτων χρειάζεται μόνο λίγα χιλιοστά του δευτερολέπτου για την ολοκλήρωση του. Θυμίζουμε επίσης ότι για τη σωστή λειτουργία του αλγορίθμου αυτού είναι απαραίτητο η επανεγγραφή εισόδου να είναι κλειστή ως προς το συμπερασμό. Παρόλα αυτά, αυτό αντικατοπτρίζεται μόνο στην επίδοση για το ερώτημα 5 των οντολογιών U και UX και για το ερώτημα 5 της AX, όπου λόγω του μεγέθους της επανεγγραφής εισόδου ο αλγόριθμος μας χρειάζεται αρκετό χρόνο για να τρέξει.

Ο Πίνακας 6.5 παρουσιάζει τη σύγκριση της Ref_E και του Requiem για το πλαίσιο που αφορά στις $\mathcal{ELHI}$ οντολογίες. Εφόσον όλες οι μεταβλητές σε όλα τα ερωτήματα για την οντολογία LUBM είναι διακεκριμένες, αφαιρέσαμε πρώτα μια διακεκριμένη μεταβλητή, στη συνέχεια υπολογίσαμε την επανεγγραφή για το ερώτημα αυτό και τέλος, τρέξαμε τον αλγόριθμο μας επεκτείνοντας το ερώτημα με την μεταβλητή που είχαμε αφαιρέσει προηγούμενα. Όπως παρατηρούμε ο αλγόριθμος μας είναι πάντα πιο

Πίνακας 6.6: Αξιολόγηση του Αλγορίθμου 5 χρησιμοποιώντας DL-Lite οντολογίες

$\mathcal{O}$	Q	Ref_L, Αλγόριθμος 5				Rapid	
		$\sharp\mathcal{R}$	t_{rew}	t_{alg}	t_{sub}	t_{R}	t_{sub}
V	2	10	0,0	2,0	1,0	20,0	1,0
	3	72	0,0	3,5	4,5	7,0	0,5
	4	185	0,0	2,0	0,5	7,0	0,0
	5	30	0,0	2,0	1,0	11,0	0,0
P5	2	10	1,0	1,0	0,0	1,0	0,0
	3	13	4,0	4,0	1,0	3,0	0,0
	4	15	6,0	6,0	1,0	5,0	1,0
	5	16	10,0	10,0	1,0	15,0	1,0
P5X	2	81	2,0	3,0	0,0	3,0	0,0
	3	413	8,0	8,0	1,0	16,0	1,0
	4	2 070	21,0	21,0	5,0	28,0	11,0
	5	10 352	211,0	211,0	81,0	159,0	120,0
S	2	204	0,0	2,5	1,0	2,5	0,0
	3	864	0,0	40,0	7,3	2,3	0,3
	4	1 428	0,0	15,0	5,7	2,0	0,0
	5	6 048	0,0	96,8	149,4	6,0	0,0
U	1	2	2,0	2,0	1,0	4,0	0,0
	2	190	0,0	1,0	0,5	1,5	0,0
	3	300	0,0	7,3	2,5	2,3	0,3
	4	1 688	0,0	16,5	0,0	1,5	0,5
	5	3 375	0,0	51,3	38,5	6,0	0,0
UX	1	5	1,0	1,0	0,0	11,0	0,0
	2	287	0,0	3,0	0,0	2,0	0,0
	3	1 260	0,0	22,5	17,8	4,8	0,0
	4	5 137	0,0	120,0	2,0	4,5	0,0
	5	8 955	0,0	152,0	88,8	14,8	0,0
A	1	378	0,0	0,0	0,0	2,5	0,5
	2	103	1,5	1,5	0,0	1,5	0,5
	3	104	2,0	3,7	4,3	10,7	0,0
	4	471	1,5	2,0	0,0	4,0	0,5
	5	624	8,7	11,7	11,7	25,3	7,0
AX	1	794	0,0	1,0	0,5	5,0	0,0
	2	1 812	1,5	2,0	1,0	6,5	0,5
	3	4 763	4,7	88,0	3 228,3	296,3	8,7
	4	7 229	2,0	5,0	9,5	27,0	5,5
	5	78 885	50,3	59,5	356,0	349,7	24,3

γρήγορος από το Requiem, με πιο αξιόλογες περιπτώσεις τα ερωτήματα στις οντολογίες της GALEN στα οποία είναι πάντα τρεις (στην G_2 ακόμα και τέσσερις) τάξεις μεγέθους γρηγορότερος. Αξίζει να σημειωθεί ότι η καλύτερη συμπεριφορά του αλγορίθμου μας σε σχέση με την πειραματική αξιολόγηση για DL-Lite οντολογίες μπορεί να δικαιολογηθεί από το μικρότερο μέγεθος των επανεγγραφών.

6.2.3 Μείωση Ερωτημάτων με Άτομα

Σε αυτή την αξιολόγηση, για κάθε ερώτημα πραγματοποιήσαμε όλες τις δυνατές αφαιρέσεις ατόμων δεδομένου ότι το παραγόμενο ερώτημα παρέμενε συνδεδεμένο και

Πίνακας 6.7: Αξιολόγηση του Αλγορίθμου 5 χρησιμοποιώντας $\mathcal{ELHI}$ οντολογίες

$\mathcal{O}$	Q	Ref_E, Αλγόριθμος 5				Requiem	
		$\sharp\mathcal{R}$	t_{rew}	t_{alg}	t_{sub}	t_{R}	t_{sub}
L	1	3	4,5	5,0	0,0	87,0	0,5
	2	30	0,0	3,5	1,8	92,2	2,8
	3	17	0,0	1,0	0,0	0,0	0,0
	4	319	0,0	0,0	1,0	56,0	1,0
	5	154	0,0	1,0	0,0	40,0	0,0
	7	1 315	0,0	10,5	28,0	44,0	60,5
	8	1 055	0,0	39,0	41,0	166,5	176,5
	9	-	-	-	-	-	-
	10	264	0,0	0,0	0,0	17,0	0,0
	11	3	0,0	0,0	0,0	0,0	0,0
	12	156	0,0	4,0	1,0	22,5	6,0
	13	185	0,0	1,0	0,0	1,0	0,0
G_1	1	16	0,0	1,0	1,0	1 150,0	1,0
	2	4 117	0,0	114,0	68,0	673,0	3,0
	3	6 742	0,0	143,0	27,0	1 416,0	82,0
	4	21	0,0	1,0	0,0	376,0	0,0
	5	111	0,0	7,0	0,0	380,0	0,0
G_2	1	80	0,0	5,0	1,0	694,0	2,0
	2	54	0,0	1,0	0,0	245,0	0,0
	3	26 859	0,0	120,0	7,0	669,0	67,0
	4	380	0,0	2,0	1,0	251,0	2,0
	5	152	0,0	0,0	0,0	169,0	1,0
G_3	2	784	62,0	67,5	25,0	1 313,0	17,5
	3	91	47,0	47,0	21,0	1 231,0	2,0
	4	2 439	23,3	67,0	8,7	1 259,3	21,3

δεν χρησιμοποιήσαμε ερωτήματα που περιείχαν μόνο ένα άτομο στο σώμα τους. Και πάλι οι χρόνοι που παρουσιάζουμε είναι μέσοι όροι.

Ο Πίνακας 6.6 παρουσιάζει τη σύγκριση ανάμεσα στην Ref_L και το Rapid για το πλαίσιο που αφορά στις DL-Lite οντολογίες. Μπορούμε να παρατηρήσουμε ότι στις περισσότερες των περιπτώσεων ο αλγόριθμος μας συμπεριφέρεται το ίδιο καλά με το Rapid (θυμίζουμε ότι το Rapid είναι ένα εξαιρετικά βελτιστοποιημένο σύστημα για την επανεγγραφή ερωτημάτων με βάση DL-Lite οντολογίες). Υπάρχουν όμως μερικές περιπτώσεις που το Rapid είναι κατά πολύ γρηγορότερο, όπως στο ερώτημα 5 για τις οντολογίες S και U, και τα ερωτήματα 4 και 5 για την UX. Αυτό οφείλεται στο γεγονός ότι η επανεγγραφή που χρησιμοποιείται ως είσοδος στον αλγόριθμο μας έχει αρκετά μεγάλο μέγεθος μιας και είναι κλειστή ως προς το συμπερασμό. Επίσης μια άλλη περίπτωση που η επίδοση του αλγορίθμου μας είναι φτωχή είναι όταν η έξοδος της Ref_L είναι μεγάλη, οπότε είναι μεγάλη και η είσοδος στην μέθοδο απαλοιφής περιττών ερωτημάτων. Κάτι τέτοιο συμβαίνει για παράδειγμα στο ερώτημα 3 της οντολογίας AX, όπου ο αλγόριθμος απαλοιφής χρειάζεται περισσότερα από 3 δευτερόλεπτα, ενώ το Rapid μόλις 8,7 χιλιοστά του δευτερολέπτου. Είναι προφανές ότι στην περίπτωση αυτή οι βελτιστοποιήσεις μας για τον έλεγχο απαλοιφής δεν είχαν καμία επίδραση.

Τέλος, ο Πίνακας 6.7 παρουσιάζει τη σύγκριση ανάμεσα στην Ref_E και στο

Requiem για το πλαίσιο που αφορά στις $\mathcal{ELHI}$ οντολογίες. Μπορούμε να παρατηρήσουμε ότι σε όλες τις περιπτώσεις η υλοποίηση μας υπερτερεί του Requiem με πιο αξιόλογες περιπτώσεις τα ερωτήματα 1, 2 και 8 από την οντολογία LUBM και όλα τα ερωτήματα από τα τμηματα της GALEN στα οποία η διαφορά μπορεί να ανέλθει σε αρκετές τάξεις μεγέθους. Αξίζει να σημειωθεί ότι, παρά το μεγάλο μέγεθος των επανεγγραφών εισόδου (ερωτήματα 2 και 3 για την G_1 και ερώτημα 3 για την G_2), η επίδοση του αλγορίθμου μας σε σχέση με το Requiem δεν επηρεάζεται μιας και το τελευταίο χρειάζεται αρκετά δευτερόλεπτα. Τέλος η υπο-ρουτίνα rew κλήθηκε το πολύ μια φορά, και για το λόγο αυτό δεν υπήρχε επιβάρυνση για τον αλγόριθμο μας.

6.3 Αξιολόγηση Επαυξητικής Επανεγγραφής

Για την πειραματική αυτή αξιολόγηση υλοποιήσαμε τους Αλγορίθμους 9–14 σε ένα σύστημα που ονομάζεται IQAROS.[5] Υλοποιήσαμε επίσης μια εκτεταμένη έκδοση του πρωτότυπου αλγόριθμου PerfectRef, που ονομάζεται PerfectRef$^+$, και χρησιμοποιεί αντί του πρωτότυπου βήματος μείωσης το βελτιστοποιημένο βήμα μείωσης που στηρίζεται στο Λήμμα 4.2.3 και στη συνάρτηση mergeCQs.

Στη συνέχεια παρουσιάζουμε τρεις διαφορετικές πειραματικές αξιολογήσεις. Η πρώτη συγκρίνει τους αλγόριθμους PerfectRef και PerfectRef$^+$ έτσι ώστε να μπορέσουμε να εκτιμήσουμε σε ποιο βαθμό το περιορισμένο βήμα μείωσης βελτιώνει την απόδοση του πρωτότυπου αλγόριθμου PerfectRef. Στη δεύτερη πειραματική αξιολόγηση συγκρίνουμε διάφορες εκδόσεις του συστήματος IQAROS, χρησιμοποιώντας κάθε φορά ένα διαφορετικό σύνολο από βελτιστοποιήσεις τις οποίες παρουσιάσαμε στην ενότητα 5.2, μεταξύ τους αλλά και με τον αλγόριθμο PerfectRef$^+$ έτσι ώστε να εκτιμήσουμε το κατά πόσον ο επαυξητικός αλγόριθμος που επεξεργάζεται βήμα-βήμα τα άτομα ενός ερωτήματος βελτιώνει την πρωτότυπη στρατηγική. Τέλος, συγκρίνουμε το IQAROS με τα πιο σύγχρονα συστήματα επανεγγραφής ερωτημάτων. Πιο συγκεκριμένα, ήμασταν σε θέση να το συγκρίνουμε με το Rapid [40], το Nyaya [64], και το Presto [132], ενώ δε το συγκρίναμε με το Requiem [116] επειδή το Rapid υπερτερεί αυτού [40]. Για την αξιολόγηση χρησιμοποιήσαμε το πλαίσιο που προτάθηκε από τους Hector et al. [116] και παρουσιάστηκε στην προηγούμενη ενότητα. Και πάλι όλοι οι χρόνοι είναι μετρημένοι σε χιλιοστά του δευτερολέπτου.

6.3.1 Συγκρίνοντας τους PerfectRef και PerfectRef$^+$

Ο Πίνακας 6.8 παρουσιάζει τα αποτελέσματα της εκτέλεσης των PerfectRef και PerfectRef$^+$ στο σύνολο των δεδομένων ελέγχου. Στον πίνακα αυτό με $\sharp u$ δηλώνουμε το μέγεθος της επανεγγραφής ΣΣΕ που έχει υπολογιστεί πριν από οποιδήποτε διαδικασία απαλοιφής περιττών ερωτημάτων και με t τον αντίστοιχο χρόνο υπολογισμού.

Από τον Πίνακα 6.8 μπορούμε να παρατηρήσουμε ότι σε πολλές περιπτώσεις, με πιο αξιόλογες τα ερωτήματα για τις οντολογίες P5, P5X και S, το μέγεθος του ΣΣΕ που παράγεται με τον PerfectRef$^+$ είναι κατά πολύ μικρότερο από αυτό που παράγεται

[5] http://code.google.com/p/iqaros/

Πίνακας 6.8: Σύγκριση μεταξύ των PerfectRef και PerfectRef$^+$

$\mathcal{O}$	Q	PerfectRef		PerfectRef$^+$		$\mathcal{O}$	Q	PerfectRef		PerfectRef$^+$		$\mathcal{O}$	Q	PerfectRef		PerfectRef$^+$	
		$\sharp u$	t	$\sharp u$	t			$\sharp u$	t	$\sharp u$	t			$\sharp u$	t	$\sharp u$	t
	1	2	0	2	1		1	6	0	6	0		1	15	5	15	4
	2	3	1	2	1		2	202	12	202	15		2	11	8	10	5
P1	3	7	1	2	0	S	3	1005	190	995	180	V	3	72	56	72	46
	4	16	5	2	0		4	1548	254	1548	247		4	185	101	185	81
	5	32	13	2	1		5	8693	8216	7855	5888		5	150	111	150	126
	1	6	1	6	1		1	2	0	2	1		1	402	24	402	28
	2	11	15	10	11		2	189	24	189	17		2	103	124	103	124
P5	3	22	256	13	152	U	3	296	112	296	112	A	3	104	656	104	677
	4	45	1828	15	875		4	1763	826	1746	808		4	492	1237	492	1264
	5	90	32255	16	5706		5	3418	2680	3410	2624		5	624	355571	624	324006
	1	14	0	14	0		1	5	0	5	1		1	783	30	783	27
	2	86	2	81	3		2	286	14	286	7		2	1812	141	1812	141
P5X	3	530	36	413	24	UX	3	1248	118	1248	121	AX	3	4763	707	4763	701
	4	3476	656	2070	325		4	5385	829	5325	818		4	7251	1282	7251	1384
	5	23744	41454	10352	6762		5	9220	2625	9200	2731		5	78885	319681	78885	337649

από τον PerfectRef. Κάτι τέτοιο είναι ιδιαίτερα εμφανές στο ερώτημα 5 στις οντολογίες P5 και P5X, το οποίο αποτελείται από μια μεγάλη αλυσίδα ρόλων της μορφής $R(x_1, x_2), \ldots, R(x_4, x_5) \rightarrow \mathcal{Q}_A(x_1)$. Συνεπώς, εφαρμόζοντας το πρωτότυπο βήμα μείωσης παρατηρούμε ότι ο PerfectRef παράγει πολλά περιττά ερωτήματα. Τα ερωτήματα αυτά είναι της μορφής $\ldots, R(x_1, x_1), \ldots \rightarrow \mathcal{Q}_A(x_1)$ και υπάγονται από τα ερωτήματα από τα οποία προκύπτουν, όπως από το $\ldots, R(x_1, y), R(y, z), \ldots \rightarrow \mathcal{Q}_A(x_1)$ στη συγκεκριμένη περίπτωση στο οποίο εφαρμόζεται το πρωτότυπο βήμα μείωσης με τον πιο γενικό ενοποιητή (mgu) $\{z \mapsto y, y \mapsto x_1\}$. Παρόλα αυτά, σε όλες τις άλλες περιπτώσεις οι διαφορές που παρουσιάζουν τα δύο συστήματα είναι αμελητέες.

Όσον αφορά στην απόδοση, εφόσον ο PerfectRef$^+$ καταφέρνει να υπολογίσει λιγότερα περιττά ερωτήματα από ότι ο PerfectRef, είναι επίσης σε αρκετές περιπτώσεις πιο αποδοτικός. Παρόλα αυτά, όταν δεν υπάρχει σημαντική διαφορά ανάμεσα στα μεγέθη των υπολογισμένων επανεγγραφών ΣΣΕ, οι PerfectRef και PerfectRef$^+$ συμπεριφέρονται παρόμοια και κάποιες φορές ο δεύτερος είναι ελαφρώς πιο αργός από τον πρώτο. Αυτό συμβαίνει λόγω του επιπλέον χρόνου που χρειάζεται για τον υπολογισμό των ελέγχων του περιορισμένου βήματος μείωσης.

6.3.2 Συγκρίνοντας το IQAROS και τον PerfectRef$^+$

Ο Πίνακας 6.9 απεικονίζει τα αποτελέσματα της σύγκρισης μας μεταξύ του αλγόριθμου PerfectRef$^+$ (που δηλώνεται στον πίνακα ως PR$^+$) και τριών διαφορετικών εκδόσεων του IQAROS. Η πρώτη (που ονομάζεται Inc$_1$) υλοποιεί τους Αλγόριθμους 11 και 13 χωρίς καθόλου βελτιστοποιήσεις. Η δεύτερη (που ονομάζεται Inc$_2$) χρησιμοποιεί τον Αλγόριθμο 14 αντί του Αλγορίθμου 11 όταν προσθέτει το τελευταίο άτομο του ερωτήματος, ενώ η τρίτη (που ονομάζεται Inc$_3$) βελτιώνει την Inc$_2$ υλοποιώντας επίσης και τις βελτιστοποιήσεις που παρουσιάστηκαν στην ενότητα 5.2 για τη βελτίωση της αποδοτικότητας του αλγορίθμου απαλοιφής περιττών ερωτημάτων. Και πάλι ο πίνακας παρουσιάζει αρχικά το μέγεθος των ΣΣΕ που έχουν υπολογιστεί, ενώ η στήλη που εμφανίζεται ως $\sharp NR$ παρουσιάζει το μέγεθος των μη-περιττών επανεγγραφών ΣΣΕ που θα έπρεπε να υπολογίσουν τα συστήματα. Αξίζει να σημειώσουμε ότι

Πίνακας 6.9: Σύγκριση του PerfectRef$^+$ και διαφόρων εκδόσεων του IQAROS

$\mathcal{O}$	Q	Μέγεθος ΣΣΕ				$\sharp NR$	Χρόνος Υπολογισμού ΣΣΕ				Χρόνος removeRedundant			
		PR+	Inc_1	Inc_2	Inc_3		PR+	Inc_1	Inc_2	Inc_3	PR+	Inc_1	Inc_2	Inc_3
V	1	15	15	15	15	15	4	7	6	6	1	1	1	1
	2	10	10	10	10	10	5	8	6	8	3	2	2	1
	3	72	72	72	72	72	46	24	12	14	93	41	37	0
	4	185	185	185	185	185	81	41	22	31	38	45	34	0
	5	150	30	30	30	30	126	11	10	17	115	3	1	13
P1	1	2	2	2	2	2	1	0	0	0	0	0	0	0
	2	2	2	2	2	2	1	1	1	1	0	0	0	0
	3	2	2	2	2	2	0	1	2	1	0	0	0	1
	4	2	2	2	2	2	0	1	3	2	0	0	0	0
	5	2	2	2	2	2	1	3	3	3	1	0	0	0
P5	1	6	6	6	6	6	1	1	1	1	0	0	1	0
	2	10	10	10	10	10	11	7	3	5	2	1	1	1
	3	13	13	13	13	13	152	76	19	19	3	2	2	1
	4	15	15	15	15	15	875	288	173	123	2	3	5	2
	5	16	16	16	16	16	5 706	838	306	362	0	3	4	2
P5X	1	14	14	14	14	14	0	0	0	0	0	0	1	1
	2	81	81	25	25	25	3	2	3	1	2	4	1	1
	3	413	413	133	103	58	24	24	6	17	16	71	13	14
	4	2 070	2 070	670	369	179	325	187	46	123	76	126	237	382
	5	10 352	10 352	3 352	1 885	718	6 762	828	214	418	1 340	1 989	864	879
S	1	6	6	6	6	6	0	0	0	1	0	0	0	0
	2	202	204	12	2	2	15	12	4	5	1	0	0	0
	3	995	864	96	4	4	180	60	8	16	5	5	1	0
	4	1 548	1 428	84	4	4	247	104	9	14	5	12	1	0
	5	7 855	6 048	672	8	8	5 888	1 018	227	160	85	128	9	0
U	1	2	2	2	2	2	1	1	1	2	0	0	0	0
	2	189	190	5	1	1	17	12	3	2	1	0	0	0
	3	296	300	20	4	4	112	77	5	8	1	2	0	0
	4	1 746	1 688	45	2	2	808	253	8	22	5	5	0	0
	5	3 410	3 375	90	10	10	2 624	527	17	41	16	55	1	0
UX	1	5	5	5	5	5	1	1	1	6	0	0	1	0
	2	286	287	7	1	1	7	10	4	7	1	1	0	0
	3	1 248	1 260	84	12	12	121	80	10	34	10	24	11	0
	4	5 325	5 137	129	5	5	818	201	11	24	19	42	4	0
	5	9 200	8 955	225	25	25	2 731	427	31	61	97	166	36	0
A	1	402	357	77	77	27	28	17	5	16	3	1	6	1
	2	103	103	54	54	50	124	39	12	10	1	0	31	0
	3	104	104	104	104	104	677	173	103	65	6	4	225	0
	4	492	471	320	320	224	1 264	170	58	57	16	27	72	1
	5	624	624	624	624	624	324 006	3 412	258	812	145	255	233	5
AX	1	783	794	431	431	41	27	18	4	7	5	6	4	3
	2	1 812	1 812	1 653	1 545	1 431	141	57	26	34	546	695	746	6
	3	4 763	4 763	4 466	4 466	4 466	701	186	48	144	7 141	9 832	7 958	47
	4	7 251	7 229	6 639	4 479	3 159	1 384	192	37	88	3 322	4 699	3 542	49
	5	78 885	78 885	74 025	32 944	32 921	337 649	4 361	665	1 559	-	-	-	840

αυτό είναι το μέγεθος του ΣΣΕ για κάθε σύστημα μετά την απαλοιφή των περιττών ερωτημάτων. Στη συνέχεια ο πίνακας παρουσιάζει τους χρόνους υπολογισμού των ΣΣΕ πριν την εφαρμογή του αλγόριθμου απαλοιφής περιττών ερωτημάτων, ενώ οι τελευταίες τέσσερις στήλες παρουσιάζουν το χρόνο που απαιτήθηκε για να εκτελεστεί ο αλγόριθμος απαλοιφής περιττών ερωτημάτων (removeRedundant) για κάθε σύστημα.

Αρχικά, παρατηρούμε ότι τα μεγέθη των ΣΣΕ που έχουν υπολογιστεί από το σύστημα Inc_1 και τον $\mathsf{PerfectRef}^+$ είναι σχεδόν πανομοιότυπα (με κάποιες μικρές διαφορές σε κάποια ερωτήματα και οντολογίες). Το γεγονός αυτό οφείλεται στο ότι και τα δύο συστήματα χρησιμοποιούν το περιορισμένο βήμα μείωσης και επειδή ο πυρήνας του το Inc_1 στηρίζεται στον ίδιο αλγόριθμο επανεγγραφής για την εξαγωγή γνώσης από το $\mathcal{T}$. Εν τούτοις, παρά τις ομοιότητές τους όσον αφορά στο ΣΣΕ που έχει υπολογιστεί, παρατηρούμε ότι το Inc_1 είναι σημαντικά πιο αποδοτικό από το $\mathsf{PerfectRef}^+$. Για παράδειγμα, στις οντολογίες P5, P5X, S, U, και UX, το Inc_1 είναι κατά πολύ ταχύτερο από το $\mathsf{PerfectRef}^+$, ενώ στο ερώτημα 5 στις οντολογίες A και AX, καταφέρνει να είναι πιο γρήγορο ακόμα και για δύο τάξεις μεγέθους από ότι το $\mathsf{PerfectRef}^+$. Εφόσον και τα δύο συστήματα στηρίζονται στην ίδια προσέγγιση για την εξαγωγή γνώσης από το $\mathcal{T}$, συμπεραίνουμε ότι η βελτίωση αυτή οφείλεται στην επαυξητική στρατηγική. Πιο συγκεκριμένα, η επαυξητική προσέγγιση παρέχει μια πιο στοχευμένη στρατηγική (επεξεργάζεται ένα άτομο τη φορά), σε σύγκριση με την τυφλή εφαρμογή των βημάτων αναδιατύπωσης και μείωσης στον $\mathsf{PerfectRef}(^+)$. Τέλος, εφόσον και τα δύο συστήματα υπολογίζουν ΣΣΕ συγκρίσιμου μεγέθους οι χρόνοι απαλοιφής των περιττών ερωτημάτων με τη χρήση του αλγόριθμου removeRedundant είναι παρόμοιοι. Αξίζει να σημειωθεί όμως, ότι ο αλγόριθμος αυτός είναι μη-ντετερμινιστικός σε μεγάλο βαθμό και συνεπώς μπορεί να προκύψουν μικρές παρεκλίσεις.

Συγκρίνοντας τις διαφορετικές εκδόσεις του IQAROS παρατηρούμε ότι το μέγεθος του ΣΣΕ που έχει υπολογιστεί μειώνεται καθώς κινούμαστε από το Inc_1 στο Inc_2 και τέλος στο Inc_3. Για παράδειγμα, σε αρκετές περιπτώσεις το Inc_2 παράγει ΣΣΕ που έχει μέγεθος ακόμα και 30 φορές μικρότερο από αυτό του Inc_1 (οντολογίες P5X, S, U, και UX). Η μείωση αυτή μπορεί να δικαιολογηθεί από το γεγονός ότι το Inc_2 χρησιμοποιεί τον (πιο απλό) Αλγόριθμο 14 αντί του Αλγόριθμου 11, όταν προσθέτει το τελευταίο άτομο του ερωτήματος. Όπως εξηγήσαμε και προηγουμένως, ο αλγόριθμος αυτός δεν υπολογίζει το καρτεσιανό γινόμενο των γράφων και επίσης δεν προσθέτει πολλά περιττά ερωτήματα (Πρόταση 5.2.1) στην τελική επανεγγραφή. Επιπρόσθετα, μπορούμε να παρατηρήσουμε ότι ο αλγόριθμος αυτός για την τελευταία επανάληψη του Αλγόριθμου 13 είναι απλοποιημένος, όπως απεικονίζεται και στους χρόνους υπολογισμού του Inc_2 σε σύγκριση με αυτούς του Inc_1. Πιο συγκεκριμένα, σχεδόν σε όλες τις οντολογίες το Inc_2 είναι αρκετές φορές πιο γρήγορο από το Inc_1. Τα αποτελέσματα του υπολογισμού μιας πολύ μικρότερης επανεγγραφής ΣΣΕ για τις οντολογίες P5X και AX φαίνονται επίσης και από το χρόνο του αλγόριθμου απαλοιφής περιττών ερωτημάτων. Εν τούτοις, αξίζει να σημειωθεί ότι κανένα από τα δύο συστήματα δεν μπορεί να υπολογίσει ένα μη-περιττό ΣΣΕ για το ερώτημα 5 της οντολογίας AX μέσα στο χρονικό όριο των 600 δευτερολέπτων.

Τέλος, το Inc_3 παράγει τη μικρότερη επανεγγραφή ΣΣΕ σε σύγκριση με όλα τα προηγούμενα συστήματα. Το γεγονός αυτό οφείλεται στις πρόσθετες τεχνικές

που έχουν υλοποιηθεί για την αναγνώριση περιττών ερωτημάτων που έχει περιγραφεί στην Πρόταση 5.2.2. Μπορούμε επίσης να παρατηρήσουμε ότι λόγω αυτών των βελτιστοποιήσεων, στις περισσότερες των περιπτώσεων το ΣΣΕ που υπολογίζεται από το Inc_3 είναι στην πραγματικότητα επίσης μη-περιττό ή περιέχει ελάχιστα περιττά ερωτήματα. Όσον αφορά τον χρόνο υπολογισμού, το Inc_3 είναι γενικά όσο γρήγορο είναι και το Inc_2, όμως, παρατηρούμε ότι σε κάποιες περιπτώσεις είναι ελαφρώς πιο αργό από το Inc_2. Αυτό μπορεί να δικαιολογηθεί από τον επιπλέον υπολογιστικό χρόνο που χρειάζονται οι διάφορες βελτιστοιποιήσεις. Εν τούτοις, όταν εξετάζουμε το χρόνο για την απαλοιφή των περιττών ερωτημάτων τα οφέλη από την ανίχνευση των (μη-)περιττών (Πρόταση 5.2.4) και μη-υπαγόντων (Πρόταση 5.2.6) ερωτημάτων γίνονται πιο εμφανή. Το Inc_3 είναι πιο γρήγορο από όλα τα άλλα συστήματα και πολύ πιο γρήγορο σε όλα τα ερωτήματα της οντολογίας AX. Μάλιστα, το Inc_3 είναι το μοναδικό σύστημα που μπορεί να υπολογίσει μια μη-περιττή επανεγγραφή ΣΣΕ για το ερώτημα 5 στην οντολογία AX. Πιο συγκεκριμένα, από τα 32.960 ερωτήματα που έχει υπολογίσει ο αλγόριθμος μετά την επεξεργασία του τελευταίου ατόμου του ερωτήματος 5 έχει αναγνωρίσει 31.593 μη-περιττά ερωτήματα (Πρόταση 5.2.4). Τα ερωτήματα αυτά παραλείπονται από τη διαδικασία απαλοιφής περιττών ερωτημάτων και συνεπώς ο αλγόριθμος τερματίζει σε λιγότερο από 2,4 δευτερόλεπτα.

6.3.3 Συγκρίνοντας τα IQAROS, Rapid, Nyaya, και Presto

Ο Πίνακας 6.10 παρουσιάζει μια σύγκριση μεταξύ του Inc_3 και των συστημάτων Rapid, Nyaya και Presto. Επιπλέον, στις στήλες που δηλώνονται ως 'Χρόνος Υπολογισμού ΣΣΕ' παρουσιάζουμε δύο επιπλέον χρόνους. Ο πρώτος, που δηλώνεται ως Inc_1^n, είναι ο χρόνος που απαιτείται από το Inc_1 για να επεξεργαστεί το τελευταίο άτομο α του ερωτήματος εισόδου $\mathcal{Q}$ χρησιμοποιώντας τον Αλγόριθμο 13, ενώ ο Inc_3^n είναι ο ίδιος χρόνος αλλά για το Inc_3 που χρησιμοποιεί τον Αλγόριθμο 14. Συνεπώς, οι χρόνοι αυτοί αντικατοπτρίζουν μόνο το χρόνο που απαιτείται για να επεκτείνουμε τον γράφο $\mathcal{G}^-$, που έχει υπολογιστεί μέχρι στιγμής για τα $\mathcal{Q}^- := \mathcal{Q} \setminus \{\alpha\}, \mathcal{T}$, σε μια επανεγγραφή για τα $\mathcal{Q}, \mathcal{T}$. Δηλαδή, εάν μας δινόταν ο $\mathcal{G}^-$ για τα $\mathcal{Q}^-, \mathcal{T}$ οι χρόνοι αυτοί αντικατοπτρίζουν μόνο το χρόνο που χρειάζεται για να επεκτείνουμε τον γράφο που δίνεται ως είσοδος σε μια επανεγγραφή ΣΣΕ για το ερώτημα $\mathcal{Q}^-$ στο οποίο προστίθεται το τελευταίο άτομο α. Εδώ πρέπει να σημειωθεί ότι η έξοδος του Inc_1^n είναι ένας γράφος επανεγγραφής, που μπορεί να επεκταθεί περαιτέρω, ενω η έξοδος του Inc_3^n είναι μια επανεγγραφή ΣΣΕ.

Όπως και προηγούμενα, μετά την εκτέλεση του αλγορίθμου απαλοιφής περιττών ερωτημάτων όλα τα συστήματα επιστρέφουν επανεγγραφές ίδιου μεγέθους (όπως αυτές περιγράφονται στον Πίνακα 6.9), εκτός από το σύστημα Presto στα ερωτήματα 2–5 για την οντολογία P5X και στα ερωτήματα 2 και 4 για την οντολογία AX. Ύστερα από ενδελεχή μελέτη, παρατηρήσαμε ότι το Presto δεν είναι πλήρες σε αυτές τις περιπτώσεις. Πιο συγκεκριμένα, αποτυγχάνει να υπολογίσει ερωτήματα που δεν υπάγονται από άλλα ερωτήματα που υπολογίζει. Συνεπώς, υπάρχει ένα σώμα ορολογίας (ABox) για το οποίο η εξίσωση 2.3 δεν ισχύει.

Σε σύγκριση με το Nyaya, το Inc_3 (καθώς και όλες οι άλλες εκδόσεις του IQAROS

Πίνακας 6.10: Σύγκριση μεταξύ Rapid, Nyaya, Presto και IQAROS

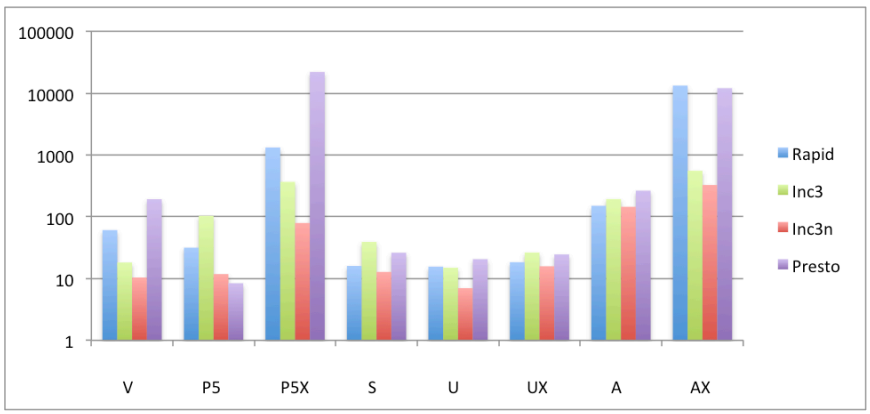
$\mathcal{O}$	Q	Μέγεθος ΣΣΕ				Χρόνος Υπολογισμού ΣΣΕ						Χρόνος removeRedundant			
		Rapid	Nyaya	Presto	Inc_3	Rapid	Nyaya	Presto	Inc_3	Inc_1^n	Inc_3^n	Rapid	Nyaya	Presto	Inc_3
V	1	15	15	15	15	13	84	793	6	6	6	0	0	1	1
	2	10	10	10	10	13	121	4	8	3	4	0	0	1	1
	3	72	72	72	72	78	360	42	14	15	9	0	0	21	0
	4	185	185	185	185	102	442	52	31	30	25	0	0	36	0
	5	30	52	30	30	98	476	12	17	5	8	0	24	2	13
P1	1	2	2	2	2	7	1	16	0	0	0	0	0	0	0
	2	2	2	2	2	9	1	1	1	1	0	0	0	0	0
	3	2	2	2	2	9	2	2	1	1	0	0	0	3	1
	4	2	2	2	2	20	6	2	2	0	1	0	0	0	0
	5	2	2	2	2	36	11	3	3	0	0	0	0	1	0
P5	1	6	6	6	6	7	14	5	1	1	1	0	0	0	0
	2	10	10	6	10	14	128	4	5	6	3	1	0	0	1
	3	13	13	6	13	22	726	16	19	62	6	1	0	1	1
	4	15	15	6	15	33	1 889	2	123	212	19	3	0	7	2
	5	16	16	6	16	75	16 062	3	362	660	30	2	0	4	2
P5X	1	14	14	14	14	10	12	20	0	0	0	0	0	1	1
	2	25	66	81	25	23	130	7	1	1	1	3	40	13	1
	3	127	374	413	103	92	540	76	17	15	13	43	875	1 095	14
	4	636	2 475	2 070	369	343	1 672	1 371	123	120	106	838	2 170	1 891	382
	5	3 180	17 584	10 352	1 885	2 061	15 095	31 797	418	581	276	3 191	127 485	73 762	879
S	1	6	6	6	6	6	15	57	1	0	1	0	0	0	0
	2	2	3	2	2	9	11	9	5	9	2	0	1	0	0
	3	4	7	4	4	14	46	23	16	45	4	0	2	0	0
	4	4	5	4	4	14	34	19	14	86	6	0	0	0	0
	5	8	13	8	8	36	159	23	160	808	51	1	4	0	0
U	1	2	2	2	2	9	25	63	2	0	1	0	0	0	0
	2	1	1	1	1	19	7	15	2	9	1	0	0	0	0
	3	4	4	4	4	13	172	14	8	68	4	1	0	0	0
	4	2	2	2	2	17	15	4	22	238	11	0	0	0	0
	5	10	11	10	10	18	107	5	41	484	18	1	1	0	0
UX	1	5	5	5	5	11	24	74	6	0	3	0	0	0	0
	2	1	1	1	1	13	6	13	7	6	3	0	0	0	0
	3	12	12	12	12	20	166	7	34	64	24	1	0	0	0
	4	5	5	5	5	17	15	23	24	160	13	0	0	0	0
	5	25	26	25	25	26	115	6	61	358	36	4	5	0	0
A	1	27	248	402	77	18	1 231	778	16	15	14	0	73	41	1
	2	54	93	103	54	43	4 928	12	10	37	6	2	39	2	0
	3	104	105	104	104	97	35 451	15	65	145	31	0	40	5	0
	4	333	455	492	320	170	17 121	47	57	153	49	38	390	65	1
	5	624	-	624	624	383	-	160	812	3 243	624	1	-	197	5
AX	1	41	556	782	431	26	1 282	2 245	7	14	6	0	367	128	3
	2	1 546	1 738	1 781	1 545	649	4 493	615	34	41	32	542	1 095	1 034	6
	3	4 466	4 742	4 752	4 466	1 694	34 032	8 000	144	129	81	531	17 320	17 260	47
	4	4 497	6 565	7 100	4 479	1 247	16 569	8 236	88	152	82	1 538	19 891	22 543	49
	5	32 956	-	-	32 944	3 810	-	-	1 559	3 628	1 430	56 196	-	-	840

από τον Πίνακα 6.9) είναι γενικά πολύ πιο γρήγορο, και σε μερικές περιπτώσεις ακόμα και για μερικές τάξεις μεγέθους. Επιπλέον, το Inc_3 υπολογίζει πολύ μικρότερα ΣΣΕ. Εφόσον το Nyaya στηρίζει κυρίως τη λειτουργία του στον ίδιο αλγόριθμο αναδιατύπωσης για την εξαγωγή γνώσης από το $\mathcal{T}$ όπως το PerfectRef οι λόγοι για τη διαφορά αυτή είναι ίδιοι με αυτούς που έχουμε αναφέρει προηγουμένως.

Σε σύγκριση με το Presto, το Inc_3 υπολογίζει μικρότερες επανεγγραφές με πιο αξιόλογες περιπτώσεις τα ερωτήματα 2–5 στην P5X, τα ερωτήματα 1 και 2 στην A και τέλος όλα τα ερωτήματα στην AX. Οι επιπτώσεις του υπολογισμού πολύ μικρότερων επανεγγραφών αντικατοπτρίζονται επίσης και στον χρόνο που απαιτείται για τον υπολογισμό της επανεγγραφής. Το Inc_3 εκτελείται γενικά πιο γρήγορα, και σε μερικές περιπτώσεις ακόμα και για μερικές τάξεις μεγέθους (ερώτημα 1 στην V, ερωτήματα 4 και 5 στην P5X, ερώτημα 1 στην A, και όλα τα ερωτήματα στην AX) με πιο αξιόλογη περίπτωση το ερώτημα 5 της AX όπου το Presto δεν καταφέρνει καν να τερματίσει στο απαιτούμενο χρονικό όριο. Παρόλα αυτά, υπάρχουν περιπτώσεις που το Presto είναι αισθητά πιο γρήγορο, όπως στο ερώτημα 5 για την οντολογία UX και το ερώτημα 5 για την A. Επιπρόσθετα μπορούμε να παρατηρήσουμε ότι η απαλοιφή περιττών ερωτημάτων για το Inc_3 είναι πολύ πιο αποδοτική από ότι στο Presto λόγω των διαφόρων τεχνικών βελτιστοποίησης που χρησιμοποιούνται για την αναγνώριση (μη-)περιττών και μη-υπαγόντων ερωτημάτων.

Σε σύγκριση με το Rapid, το Inc_3 υπολογίζει ομοίως μικρές επανεγγραφές ΣΣΕ, με λίγες μικρές εξαιρέσεις (είτε υπέρ είτε κατά) στα ερωτήματα 3–5 στην οντολογία P5X, στο ερώτημα 1 στις οντολογίες A και AX και στα ερωτήματα 2, 4 και 5 στην οντολογία AX. Επιπρόσθετα, το Rapid είναι αισθητά γρηγορότερο μόνο στα ερωτήματα 4 και 5 στην P5 και στο ερώτημα 5 στις οντολογίες S και A. Εν τούτοις, ακόμα και σε αυτές τις περιπτώσεις η διαφορά τους είναι αρκετά μικρή καθώς δεν ξεπερνάει ποτέ τα 253 χιλιοστά του δευτερολέπτου. Σε όλες τις άλλες περιπτώσεις ο Inc_3 είναι πιο γρήγορος με πιο σημαντικές περιπτώσεις τα ερωτήματα 4 και 5 στην P5X και τα 2–5 στην AX. Επιπλέον, παρατηρούμε ότι ο αλγόριθμος απαλοιφής περιττών ερωτημάτων για τον Inc_3 είναι πολύ πιο αποδοτικός από αυτόν του Rapid με πιο σημαντική περίπτωση πάλι το ερώτημα 5 στην οντολογία AX. Για ακόμα μια φορά, οι διαφορές αυτές δικαιολογούνται από τις τεχνικές βελτιστοποίησης που χρησιμοποιούνται στον Inc_3 έτσι ώστε να μπορεί να αναγνωρίσει (μη-)περιττά και μη-υπάγοντα ερωτήματα.

Παρόλα αυτά, παρατηρούμε ότι η πιο γρήγορη προσέγγιση είναι αυτή του Inc_3^n. Συνεπώς, ο υπολογισμός μιας επανεγγραφής ΣΣΕ επεκτείνοντας έναν γράφο επανεγγραφής (που έχει υπολογιστεί προηγουμένως) είναι όντως η πιο γρήγορη μέθοδος για ένα ερώτημα που έχει επεκταθεί. Όμως, όπως έχει αναφερθεί και προηγουμένως, η έξοδος του αλγορίθμου αυτού δεν είναι ένας γράφος επανεγγραφής και συνεπώς δεν μπορεί να χρησιμοποιηθεί για περαιτέρω επεκτάσεις του ερωτήματος. Εν τούτοις, παρατηρώντας τον χρόνο υπολογισμού του Inc_1^n βλέπουμε ότι ένας γράφος επανεγγραφής μπορεί να υπολογιστεί επίσης σχετικά αποδοτικά. Άρα, υποστηρίζουμε ότι εάν ένα ερώτημα $\mathcal{Q}$ επεκταθεί με ένα άτομο α, ο Αλγόριθμος 14 μπορεί να χρησιμοποιηθεί για να υπολογίσουμε αποδοτικά μια νέα επανεγγραφή ΣΣΕ για το $\mathcal{Q}' := \bigwedge \mathsf{bd}(\mathcal{Q}) \cup \{\alpha\} \rightarrow \mathsf{hd}(\mathcal{Q})$, ενώ ταυτόχρονα, ως μια διαδικασία που εκτελείται στο παρασκήνιο, ο Αλγόριθμος 13 μπορεί να χρησιμοποιηθεί για τον υπολογισμό



Σχήμα 6.1: Μέσος χρόνος επανεγγραφής για όλα τα ερωτήματα για κάθε οντολογία.

ενός νέου γράφου επανεγγραφής για το $\mathcal{Q}'$, που να χρησιμοποιηθεί στη συνέχεια με παρόμοιο τρόπο για τον υπολογισμό μιας επανεγγραφής ΣΣΕ και ενός γράφου επανεγγραφής για επεκτάσεις του $\mathcal{Q}'$.

Ολοκληρώνοντας την πειραματική μας αξιολόγηση, το Σχήμα 6.1 παρουσιάζει (χρησιμοποιώντας λογαριθμική κλίμακα) τον μέσο χρόνο υπολογισμού (συνολικά για τον χρόνο επανεγγραφής και για την τελική απαλοιφή ερωτημάτων) για κάθε οντολογία, ερώτημα και σύστημα που εμφανίζεται στον Πίνακα 6.10. Τα αποτελέσματα που φαίνονται στο σχήμα αυτό επιβεβαιώνουν την προηγούμενη μας ανάλυση—δηλαδή ότι, στις οντολογίες V, P5X και AX, παρατηρούμε ότι το Inc_3^n είναι το πιο αποδοτικό σύστημα ακολουθούμενο από το Inc_3, το Rapid και στη συνέχεια το Presto. Στις περισσότερες όμως από τις υπόλοιπες οντολογίες όλα τα συστήματα παρουσιάζουν παρόμοια απόδοση και δεν παρατηρούνται μεγάλες διαφορές.

Τέλος, πραγματοποιήσαμε μια σύντομη ανάλυση για τη μνήμη που χρειάστηκε κάθε σύστημα ώστε να υπολογίσει την τελική επανεγγραφή. Η μέγιστη μνήμη που απαιτήθηκε από το Rapid είναι 140MB, ακολουθούμενο από το Nyaya με 150MB, το Presto με 180MB και τέλος το Inc_3 με 200MB. Οι τιμές αυτές είναι οι μέγιστες τιμές για κάθε σύστημα και παρουσιάστηκαν στο ερώτημα 5 για την οντολογία AX.

Μέρος IV

Απάντηση ερωτημάτων πάνω από ένα Δίκτυο Οντολογιών

Κεφάλαιο 7

Απάντηση Ερωτημάτων πάνω από ένα Δίκτυο Οντολογιών

Στα προηγούμενα κεφάλαια δείξαμε πώς μπορούμε να κατασκευάσουμε αποδοτικά μια επανεγγραφή, για σταθερά ερωτήματα αλλά και για ερωτήματα τα οποία έχουν τροποποιηθεί με διάφορους τρόπους, με βάση μια οντολογία. Στο κεφάλαιο αυτό επεκτείνουμε το πρόβλημα της απάντησης ερωτημάτων με βάση μια οντολογία σε αυτό της απάντησης ερωτημάτων πάνω από ένα δίκτυο οντολογιών. Για να είναι εφικτό κάτι τέτοιο είναι απαραίτητο να υπάρχει σημασιολογική διαλειτουργικότητα ανάμεσα στις συμμετέχουσες οντολογίες και για το λόγο αυτό είναι σκόπιμο να χρησιμοποιηθεί ένας αλγόριθμος αντιστοίχισης οντολογιών. Παρόλα αυτά οι περισσότεροι αλγόριθμοι αντιστοίχισης δεν λαμβάνουν πάντα υπ' όψιν τη σημασιολογία των οντολογιών που συμμετέχουν, με αποτέλεσμα να κατασκευάζουν αντιστοιχίσεις οι οποίες οδηγούν σε ασυνέπειες. Για το λόγο αυτό χρησιμοποιούμε μια προσέγγιση στην οποία ερμηνεύουμε τις αντιστοιχίσεις μέσω της Ασαφούς Συνολοθεωρίας και της Ασαφούς Λογικής [168] κατασκευάζοντας έτσι από ένα δίκτυο οντολογιών μια ασαφή οντολογία [153, 152]. Στη συνέχεια εξετάζουμε την παραγόμενη οντολογία για ασυνέπειες και τροποποιούμε ανάλογα τα βάρη των αντιστοιχίσεων μας ώστε να τις επιλύσουμε. Έχοντας τέλος επιλύσει όλες τις πιθανές ασυνέπειες παρουσιάζουμε ένα αλγόριθμο απάντησης ερωτημάτων.

Αρχικά στην ενότητα 7.1 παρουσιάζουμε μια εισαγωγή στην επέκταση της ΠΛ DL-Lite με την Ασαφή Συνολοθεωρία και την Ασαφή Λογική που ονομάζεται f-DL-Lite. Στη συνέχεια, στην ενότητα 7.2 παρουσιάζουμε το πρόβλημα της αντιστοίχισης οντολογιών και πώς μπορούμε να χρησιμοποιήσουμε την ασαφή συνολοθεωρία και την ασαφή λογική έτσι ώστε να παρέχουμε σημασιολογία για τις αντιστοιχίσεις. Ακολούθως, στην ενότητα 7.3 παρουσιάζουμε το πρόβλημα της απάντησης ερωτημάτων πάνω από ένα δίκτυο οντολογιών παρουσιάζοντας ταυτόχρονα προβλήματα ασυνέπειας που μπορεί να προκύψουν για τις οντολογίες μας. Για το λόγο αυτό, στην ενότητα 7.3.1 παρουσιάζουμε έναν αλγόριθμο επικύρωσης αντιστοιχίσεων και τέλος, στην ενότητα 7.3.2 παρουσιάζουμε τον αλγόριθμο μας για την απάντηση ερωτημάτων πάνω από ένα δίκτυο οντολογιών.

7.1 Ασαφείς Περιγραφικές Λογικές

Στην ενότητα αυτή, παρέχουμε μια εισαγωγή στην ασαφή ΠΛ f-DL-Lite [158]. Όπως και στις κλασικές ΠΛ μια ασαφής ΠΛ ορίζεται από ένα αλφάβητο από διαφορετικά σύνολα από ατομικές έννοιες $\mathbf{C}$, ατομικούς ρόλους $\mathbf{R}$ και άτομα $\mathbf{I}$. Όμοια με τον Ορισμό 2.4.1 οι f-DL-Lite ρόλοι κατασκευάζονται με βάση τους παρακάτω κανόνες σύνταξης όπου R ένας *βασικός ρόλος* (*basic role*) και $P \in \mathbf{R}$:

$$R ::= P \mid P^-$$

Και πάλι ένας βασικός ρόλος της μορφής P^- ονομάζεται *αντίστροφος* (*inverse*) ρόλος του P.

Οι f-DL-Lite$_{core}$-έννοιες κατασκευάζονται, όμοια, με βάση τον παρακάτω κανόνα σύνταξης, όπου B μια *βασική έννοια* (*basic concept*), C μια *γενική έννοια* (*general concept*), $A \in \mathbf{C}$ και R ένας βασικός ρόλος:

$$B ::= A \mid \exists R.\top \qquad C ::= B \mid \neg B$$

Μια ασαφής οντολογία $\mathcal{O}$ είναι ένα ζευγάρι $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$, όπου $\mathcal{T}$ είναι ένα TBox, και $\mathcal{A}$ ένα Abox. Το TBox αποτελείται από αξιώματα υπαγωγής εννοιών της μορφής $B \sqsubseteq C$, όπου οι B, C είναι f-DL-Lite-έννοιες. Το ABox από την άλλη μεριά είναι ένα σύνολο *ασαφών αναθέσεων εννοιών* (*fuzzy concept assertion*) και *ασαφών αναθέσεων ρόλων* (*fuzzy role assertions*) της μορφής $(a : C)\bowtie n$ και $((a, b) : R)\bowtie n$, ή σχέσεις ισότητας και ανισότητας ατόμων της μορφής $a = b$ ή $a \neq b$, με $a, b \in \mathbf{I}$, $\bowtie \in \{\geq, >, \leq, <\}$ και $n \in [0, 1]$.

Παρόλο που ο ορισμός των DL-Lite-εννοιών και ρόλων είναι ίδιος με αυτόν των f-DL-Lite-εννοιών και ρόλων η σημασιολογία της γλώσσας f-DL-Lite είναι αρκετά διαφορετική. Αυτό συμβαίνει γιατί με τη γλώσσα αυτή αποδίδεται μια ασαφής ερμηνεία στα δομικά σημεία της γλώσσας, όπως είναι οι έννοιες, οι ρόλοι και οι κατασκευαστές. Για το λόγο αυτό η σημασιολογία της ασαφούς γλώσσας f-DL-Lite ορίζεται με τη βοήθεια των *ασαφών ερμηνειών* (*fuzzy interpretations*) [157]. Μια ασαφής ερμηνεία $\mathcal{I}$ είναι ένα ζευγάρι $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, όπου ο χώρος ερμηνείας $\Delta^{\mathcal{I}}$ είναι, όπως και στις κλασικές ΠΛ, ένα μη-κενό σύνολο από αντικείμενα και $\cdot^{\mathcal{I}}$ είναι μια *ασαφής συνάρτηση ερμηνείας* (*fuzzy interpretation function*), η οποία απεικονίζει

- ένα άτομο $a \in \mathbf{I}$ σε ένα στοιχείο $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$,
- μια ατομική έννοια $A \in \mathbf{C}$ σε μια συνάρτηση συμμετοχής $A^{\mathcal{I}} : \Delta^{\mathcal{I}} \to [0, 1]$ και
- έναν ατομικό ρόλο $P \in \mathbf{R}$ σε μια συνάρτηση συμμετοχής της μορφής $P^{\mathcal{I}} : \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \to [0, 1]$.

Διαισθητικά ένα αντικείμενο (ζεύγος αντικειμένων) μπορεί να ανήκει σε μια έννοια (ρόλο) σε οποιοδήποτε βαθμό ανάμεσα στο 0 και 1. Για παράδειγμα η σχέση $\text{Ψηλός}^{\mathcal{I}}(\text{Γιώργος}^{\mathcal{I}}) = 0.7$, σημαίνει ότι ο Γιώργος είναι ψηλός με βαθμό ίσο με 0.7.

Οι σύνθετες f-DL-Lite-έννοιες, ρόλοι και αξιώματα ερμηνεύονται επεκτείνοντας την ασαφή ερμηνεία χρησιμοποιώντας τελεστές και ορολογία από τη ασαφή συνολοθεωρία. Η πλήρης σημασιολογία παρουσιάζεται στον Πίνακα 7.1 όπου sup είναι το

Πίνακας 7.1: Σημασιολογία f-DL-Lite-εννοιών και ρόλων

Κατασκευαστής	**Σύνταξη**	**Σημασιολογία**
καθολική έννοια	$\top$	$\top^{\mathcal{I}}(a) = 1$
σύζευξη	$B_1 \sqcap B_2$	$(B \sqcap C)^{\mathcal{I}}(a) = t(B_1^{\mathcal{I}}(a), B_2^{\mathcal{I}}(a))$
άρνηση	$\neg B$	$(\neg B)^{I}(a) = c(B^{\mathcal{I}}(a))$
υπαρξιακός περιορισμός	$\exists R.\top$	$(\exists R.\top)^{\mathcal{I}}(a) = \sup_{b \in \Delta^{\mathcal{I}}}(R^{\mathcal{I}}(a,b))$
υποέννοια	$B \sqsubseteq C$	$B^{\mathcal{I}}(a) \leq C^{\mathcal{I}}(a)$
στιγμιότυπο έννοιας	$o : C \bowtie n$	$C^{\mathcal{I}}(o^{\mathcal{I}}) \bowtie n$
στιγμιότυπο ρόλου	$(o, o') : R \bowtie n$	$R^{\mathcal{I}}(o^{\mathcal{I}}, o'^{\mathcal{I}}) \bowtie n$
ξένες έννοιες	$B_1 \sqsubseteq \neg B_2$	$B_1^{\mathcal{I}}(a) \leq 1 - B_2^{\mathcal{I}}(a)$
αντίστροφος ρόλος	P^-	$(P^-)^{\mathcal{I}}(b,a) = P^{\mathcal{I}}(a,b)$

supremum, inf είναι το *infimum*, c είναι το *ασαφές συμπλήρωμα* (*fuzzy complement*), t είναι μια *ασαφής τομή* (*fuzzy conjunction*) ή απλά τ-νόρμα (t-norm),

Μια ασαφής οντολογία $\mathcal{O}$ είναι ικανοποιήσιμη ανν υπάρχει μια ασαφής ερμηνεία $\mathcal{I}$ που να ικανοποιεί όλα τα αξιώματα στην $\mathcal{O}$. Βασικά προβλήματα συλλογιστικής στην f-DL-Lite είναι:

1. ο έλεγχος εάν μια οντολογία είναι *συνεπής*, εάν δηλαδή έχει τουλάχιστον ένα μοντέλο,

2. ο έλεγχος εάν η έννοια C *υπάγει* την B μ.β.τ. $\mathcal{O}$, εάν δηλαδή $\mathcal{O} \models B \sqsubseteq C$,

3. ο έλεγχος εάν το a είναι *στιγμιότυπο* της έννοιας C με βαθμό $\bowtie n$, εάν δηλαδή $\mathcal{O} \models a : C \bowtie n$, όπου $\bowtie \in \{\geq, >, \leq, <\}$, και

4. ο καθορισμός του *μεγιστου κατώτερου όριου* (*greatest lower bound*) του a μ.β.τ. $\mathcal{O}$, που συμβολίζεται με $glb(\Sigma, a)$, όπου $glb(\Sigma, a) = \sup\{n \mid \Sigma \models a \geq n\}$.

7.1.1 Απάντηση Ερωτημάτων

Ένα άλλο πρόβλημα συλλογιστικής στην f-DL-Lite είναι η απάντηση ερωτημάτων. Για το λόγο αυτό σε αυτή την ενότητα επεκτείνουμε τις ασαφείς ερμηνείες για τα ΣΕ. Θεωρούμε ένα ΣΕ μια πρόταση της μορφής

$$\exists \vec{y}.\phi(\vec{x}, \vec{y}) \rightarrow \mathcal{Q}_A(\vec{x})$$

ακολουθώντας τον ορισμό που έχουμε δώσει στην ενότητα 2.3. Η διάζευξη ατόμων ϕ ερμηνεύεται με τη χρήση της τ-νόρμας, ενώ ο υπαρξιακός ποσοδείκτης ερμηνεύεται με τη χρήση του sup. Σύνεπώς για $\mathbf{c}$,$\mathbf{c}' \in \Delta \times \ldots \times \Delta$ ο βαθμός της $\exists \vec{y}.\phi(\vec{x}, \vec{y})$ μέσω της $\mathcal{I}$ είναι το $\sup_{\mathbf{c}' \in \Delta}\{\phi(\mathbf{c}, \mathbf{c}')^{\mathcal{I}}\}$ για κάθε $\mathbf{c} \in \Delta \times \ldots \times \Delta$ και συνεπώς

$$Q_A^{\mathcal{I}}(\mathbf{c}) = \sup_{\mathbf{c}' \in \Delta}\{\phi(\mathbf{c}, \mathbf{c}')^{\mathcal{I}}\}.$$

Τότε λέμε ότι μια f-DL-Lite οντολογία $\mathcal{O} = \langle \mathcal{T}, \mathcal{A} \rangle$ συνεπάγεται το $\mathcal{Q}_A(\mathbf{c})$ με βαθμό n και γράφεται $\mathcal{O} \models \langle Q_A(\mathbf{c}), n \rangle$ ανν για κάθε μοντέλο $\mathcal{I}$ της $\mathcal{O}$ ισχύει $Q_A^{\mathcal{I}}(\mathbf{c}) \geq n$.

7.2 Αντιστοίχιση Οντολογιών και Ασαφής Ερμηνεία Αντιστοιχίσεων

Όπως αναφέραμε και στην εισαγωγή οι οντολογίες πολλές φορές μπορεί να είναι ετερογενείς. Για παράδειγμα δύο οντολογίες που περιγράφουν παρόμοια πεδία ενδιαφέροντιος μπορεί να ορίζουν την ίδια πραγματική οντότητα με διαφορετικούς τρόπους, ενώ είναι συνηθισμένο και το φαινόμενο κατά το οποίο οι δύο αυτές οντολογίες μπορεί να περιγράφονται με διαφορετική εκφραστικότητα. Για να μπορέσουν όμως οι οντολογίες να χρησιμοποιηθούν στον Σημασιολογικό Ιστό, είναι απαραίτητη η σημασιολογική τους διαλειτουργικότητα. Για να επιτευχθεί κάτι τέτοιο είναι απαραίτητο οι ετερογενείς οντολογίες που περιγράφουν επικαλυπτόμενα πεδία ενδιαφέροντος να αντιστοιχιστούν με ένα (ημι-)αυτόματο τρόπο, έτσι ώστε να προκύψουν αντιστοιχίσεις ανέμεσα στα στοιχεία τους [46, 164, 83, 35, 3, 51, 159, 56, 47, 97, 44]. Παρόλα αυτά, τα περισσότερα συστήματα αντιστοίχισης οντολογιών δεν λαμβάνουν υπ' όψιν τη σημασιολογία των οντολογιών, και συνεπώς οι αντιστοιχίσεις που παράγουν δεν μπορούν να χρησιμοποιηθούν ως σημασιολογικές σχέσεις, κάτι το οποίο είναι απαραίτητο έτσι ώστε να επιτευχθεί διαλειτουργικότητα ανάμεσα στις οντολογίες και κατ' επέκταση να είναι εφικτή και η απάντηση ερωτημάτων πάνω από ένα δίκτυο οντολογιών.

Για να μπορέσουμε να αποδώσουμε μια ασαφή ερμηνεία στις αντιστοιχίσεις είναι απαραίτητο να παρουσιάσουμε πρώτα τον κλασικό ορισμό τους.

Ορισμός 7.2.1. Έστω $\mathcal{O}, \mathcal{O}'$ οντολογίες, με οντότητες C_i και C'_i αντίστοιχα, έστω n_i τιμή που αποτελεί μέρος της δομής $\langle \mathcal{D}, \leq, 0, 1\rangle$, με $\mathcal{D}$ σύνολο βαθμών τ.ω. $\forall d \in \mathcal{D}, 0 \leq d \leq 1$ και έστω R_i μια από τις σχέσεις $R = \{\equiv, \sqsubseteq, \sqsupseteq\}$. Τότε, μια αντιστοίχιση είναι μια πλειάδα της μορφής

$$m_i = \langle C_i, C'_i, n_i, R_i\rangle$$

που δηλώνει ότι ισχύει η σχέση R_i με τιμή n_i για τις οντότητες C_i και C'_i. Επιπλέον μια αντιστοίχιση από την οντολογία $\mathcal{O}$ στην $\mathcal{O}'$, έστω $\mathcal{M}$, είναι ένα σύνολο αντιστοιχίσεων m_i. △

Ένας άλλος τρόπος για να ορίσουμε τις σχέσεις αυτές είναι χρησιμοποιώντας τους *κανόνες γεφύρωσης* (*bridge rules*), όπως αυτοί χρησιμοποιούνται στις *Κατανεμημένες Περιγραφικές Λογικές* (*Distributed Description Logics*) [23] και είναι εκφράσεις της μορφής

$$C_i \xrightarrow{\equiv} C'_i : n \qquad C_i \xrightarrow{\sqsubseteq} C'_i : n \qquad C_i \xrightarrow{\sqsupseteq} C'_i : n$$

Για να μπορέσουμε να λάβουμε υπόψιν την ασαφή και αβέβαιη φύση των αντστοιχήσεων στη συνέχεια ορίζουμε την ασαφή αντιστοίχιση.

Ορισμός 7.2.2. Έστω C_i και C'_i οντότητες. Μια ασασφής αντιστοίχιση $fm_i = \langle C_i, C'_i, n_i, R_i\rangle$ είναι μια αντιστοίχιση m_i, η τιμή n_i της οποίας ορίζει το βαθμό για τον οποίο ισχύει η σχέση R_i ανάμεσα στα C_i και C'_i, όπου η R_i μπορεί να είναι μια από τις σχέσεις ισοδυναμίας ($\equiv$) ή υπαγωγής ($\sqsubseteq$, $\sqsupseteq$). △

Με τον τρόπο αυτό οι αντιστοιχίσεις ορίζονται τυπικά ως ασαφής γνώση. Η βασική ιδέα πίσω από αυτό τον ορισμό είναι πως οι αντιστοιχίσεις μπορούν να χρησιμοποιηθούν έτσι ώστε να κατασκευαστούν ασαφείς ισχυρισμοί (αναθέσεις). Για να πραγματοποιηθεί κάτι τέτοιο θα πρέπει να ορίσουμε σημασιολογία για τις αντιστοιχίσεις και για να το κάνουμε αυτό θα χρησιμοποιήσουμε την Ασαφή Συνολοθεωρία [168].

Ορισμός 7.2.3. Έστω C_i, C'_i έννοιες. Έστω $\mathcal{I}$ ασαφής ερμηνεία, και έστω $\mathcal{I}_c$ μια ερμηνεία για τις κλασικές ΠΛ. Τότε θα ισχύουν οι ακόλουθες συνθήκες:

$$\begin{aligned}
\mathcal{I} \models C_i \xrightarrow{\equiv} C'_i : n_i &\iff \forall b.b \in C_i^{\mathcal{I}_c} \rightarrow C_i'^{\mathcal{I}}(b) = n_i \\
\mathcal{I} \models C_i \xrightarrow{\sqsubseteq} C'_i : n_i &\iff \forall b.b \in C_i^{\mathcal{I}_c} \rightarrow C_i'^{\mathcal{I}}(b) \geq n_i \\
\mathcal{I} \models C_i \xrightarrow{\sqsupseteq} C'_i : n_i &\iff \forall b.b \in C_i^{\mathcal{I}_c} \rightarrow C_i'^{\mathcal{I}}(b) \leq n_i
\end{aligned}$$

△

Οι παραπάνω ορισμοί συνεπάγονται μια διαδικασία μέσω της οποίας μπορούμε να μεταφέρουμε άτομα από μια οντολογία πηγής $\mathcal{O}_1$ σε μια οντολογία στόχο $\mathcal{O}_2$, κατασκευάζοντας έτσι ένα σύνολο από ασαφείς αναθέσεις A_M. Η διαδικασία αυτή παρουσιάζεται πιο τυπικά στον παρακάτω ορισμό.

Ορισμός 7.2.4. Έστω $O_1 = \langle \mathcal{T}_1, \mathcal{A}_1 \rangle$ και $O_2 = \langle \mathcal{T}_2, \mathcal{A}_2 \rangle$ οντολογίες, και έστω $\mathcal{M}$ σύνολο από αντιστοιχίσεις της μορφής $m_i = \{C_i, C'_i, n_i, R_i\} \in \mathcal{M}$. Τότε από την αντιστοίχιση των οντολογιών $\mathcal{O}_1$ και $\mathcal{O}_2$ κατασκευάζεται μια οντολογία $\mathcal{O}' = \langle \mathcal{T}', \mathcal{A}' \rangle$ τέτοια ώστε $\mathcal{T}' = \mathcal{T}_2$ και $\mathcal{A}' = \mathcal{A}_2 \cup \mathcal{A}_M$, όπου το $\mathcal{A}_M$ ορίζεται ως εξής:

$$\begin{aligned}
\mathcal{A}_M \quad = \quad & \{a : C'_i \geq n_i \mid \langle C_i, C'_i, n_i, \sqsubseteq \rangle \in M, O_1 \models C_i(a)\} \cup \\
& \{a : C'_i = n_i \mid \langle C_i, C'_i, n_i, = \rangle \in M, \mathcal{O}_1 \models C_i(a)\} \cup \\
& \{a : C'_i \leq n_i \mid \langle C_i, C'_i, n_i, \sqsupseteq \rangle \in M, \mathcal{O}_1 \models C_i(a)\}.
\end{aligned}$$

△

Όπως φαίνεται από τον παραπάνω ορισμό για την κατασκευή του $\mathcal{A}_M$ λαμβάνονται υπ' όψιν τόσο οι ισχυρισμοί που δηλώνονται ρητά όσο και αυτοί που προκύπτουν από διαδικασίες συλλογιστικής ($O_1 \models C_i(a)$). Για να είναι αυτό εφικτό είναι απαραίτητη η χρήση μιας μηχανής συλλογιστικής για κλασικές ΠΛ [146, 162, 71, 20, 59, 106]. Για παράδειγμα, εάν $m_i = \langle C_i, C'_i, 0.8, \equiv \rangle$ και $\mathcal{O}_1 \models C_i(a)$ τότε $\mathcal{A}_M = \mathcal{A}_M \cup \{a : C'_i = 0.8\}$.

7.3 Απάντηση Ερωτημάτων μέσω Αντιστοιχίσεων

Στην ενότητα αυτή μελετάμε το πρόβλημα της απάντησης ερωτημάτων μέσω αντιστοιχίσεων. Για το λόγο αυτό παρουσιάζουμε το παρακάτω παράδειγμα που θα μας καθοδηγήσει στην κατασκευή του αλγορίθμου μας.

Παράδειγμα 7.3.1. Έστω δύο οντολογίες $\mathcal{O}_1 = \langle \mathcal{T}_1, \mathcal{A}_1 \rangle$ και $\mathcal{O}_2 = \langle \mathcal{T}_2, \mathcal{A}_2 \rangle$, που περιέχουν τα παρακάτω αξιώματα:

$$\mathcal{T}_1 : \quad \mathsf{ΚινητόΤηλέφωνο} \sqsubseteq \mathsf{ΚινητήΣυσκευή}$$

$$\begin{array}{lll} \mathcal{T}_2 : & \mathsf{Τηλέφωνο} & \sqsubseteq \mathsf{ΗλεκτρονικήΣυσκευή} \\ & \mathsf{ΣταθΤηλέφωνο} & \sqsubseteq \mathsf{Τηλέφωνο} \\ & \mathsf{ΚινΤηλέφωνο} & \sqsubseteq \mathsf{Τηλέφωνο} \\ & \mathsf{ΣταθΤηλέφωνο} & \sqsubseteq \neg\mathsf{ΚινΤηλέφωνο} \end{array}$$

και τους ισχυρισμούς $\mathcal{A}_1 = \{\mathsf{ΚινητόΤηλέφωνο}(\mathsf{κτ}_1), \mathsf{ΚινητήΣυσκευή}(\mathsf{κσ}_1)\}$. Επιπλέον, έστω ότι σε κάθε έννοια από την οντολογία $\mathcal{O}_2$ ανήκει τουλάχιστον ένα άτομο. Τέλος, έστω το ερώτημα

$$\mathcal{Q} = \mathsf{ΗλεκτρονικήΣυσκευή}(x) \rightarrow \mathcal{Q}_A(x)$$

στην $\mathcal{O}_2$ οι απαντήσεις του οποίου είναι όλα τα άτομα που ανήκουν στις έννοιες της οντολογίας $\mathcal{O}_2$, καθώς όλες οι έννοιες της οντολογίας υπάγονται στην έννοια ΗλεκτρονικήΣυσκευή.

Έστω, τώρα ότι οι $\mathcal{O}_1$ και $\mathcal{O}_2$ έχουν αντιστοιχιστεί και έστω ότι το αποτέλεσμα της διαδικασίας αυτής είναι το παρακάτω σύνολο αντιστοιχίσεων:

1. map(ΚινητήΣυσκευή, ΗλεκτρονικήΣυσκευή, 0.7)
2. map(ΚινητόΤηλέφωνο, Τηλέφωνο, 0.6)
3. map(ΚινητόΤηλέφωνο, ΣταθΤηλέφωνο, 0.4)
4. map(ΚινητόΤηλέφωνο, ΚινΤηλέφωνο, 1.0)

Τότε, όπως έχουμε παρουσιάσει στον Ορισμό 7.2.4 μέσω των αντιστοιχίσεων που έχουμε προκύπτουν νέοι ασαφείς ισχυρισμοί για τις έννοιες της $\mathcal{O}_2$. Χρησιμοποιώντας πρώτα τις αντιστοιχίσεις με μεγαλύτερο βαθμό, θεωρώντας πιο σημαντικές τις αντιστοιχίσεις που παρουσιάζουν μεγαλύτερη ομοιότητα, έχουμε ότι λόγω της αντιστοίχισης 4 προστίθεται στην $\mathcal{O}_2$ ο ισχυρισμός (ΚινΤηλέφωνο($\mathsf{κτ}_1$), 1.0). Επιπρόσθετα, λόγω της αντιστοίχισης 1 έχουμε την προσθήκη των ισχυρισμών (ΗλεκτρονικήΣυσκευή($\mathsf{κσ}_1$), 0.7) και (ΗλεκτρονικήΣυσκευή($\mathsf{κτ}_1$), 0.7). Η προσθήκη όμως του τελευταίου ισχυρισμού παραβιάζει την ασαφή ερμηνεία της υπαγωγής ($C \sqsubseteq D \Longleftrightarrow C^{\mathcal{I}}(a) \leq D^{\mathcal{I}}(a)$), και συνεπώς κάνει την παραγόμενη οντολογία ασυνεπή. Επιπλέον ασυνέπειες προκύπτουν και από την προσθήκη ισχυρισμών λόγω της αντιστοίχισης 2, εφόσον παραβιάζεται και πάλι η ασαφής ερμηνεία της υπαγωγής, αλλά και της αντιστοίχισης 3. Από την τελευταία κατασκευάζεται ο ισχυρισμός (ΣταθΤηλέφωνο($\mathsf{κτ}_1$), 0.4) που παραβιάζει τον σημασιολογικό περιορισμό $\mathsf{ΣταθΤηλέφωνο}^{\mathcal{I}}(\mathsf{κτ}_1^{\mathcal{I}}) \leq 1 - \mathsf{ΚινΤηλέφωνο}^{\mathcal{I}}(\mathsf{κτ}_1^{\mathcal{I}})$. ♢

Το πρόβλημα με το παραπάνω παράδειγμα είναι πως με την προσθήκη κάποιων ασαφών ισχυρισμών στην οντολογία $\mathcal{O}_2$, η οντολογία μπορεί να γίνει ασυνεπής με αποτέλεσμα κάθε ερώτημα που τίθεται να έχει άπειρες απαντήσεις, εφόσον από μια ασυνεπή οντολογία είναι δυνατόν να προκύψει οποιοσδήποτε ισχυρισμός. Για να αποφευχθεί

Αλγόριθμος 15 fuzzyValidation($\mathcal{M}, \mathcal{O}_1, \mathcal{O}_2$)

Είσοδος: Δύο οντολογίες $\mathcal{O}_1, \mathcal{O}_2$ και ένα σύνολο αντιστοιχίσεων $\mathcal{M}$ ανάμεσα στις $\mathcal{O}_1, \mathcal{O}_2$

1: **repeat**
2: φθίνουσα ταξινόμηση του $\mathcal{M}$ μ.β.τ. βαθμό n_i των $m_i = \{C_i, C'_i, n_i, R_i\} \in \mathcal{M}$
3: $\mathcal{M}' := \emptyset$
4: **for** $m_i \in \mathcal{M}$ **do**
5: $n'_i :=$ computeStrength($m_i, \mathcal{O}_2$)
6: **if** $n'_i \neq n_i$ **then**
7: $m_i := \{C_i, C'_i, n'_i, R_i\}$
8: **break**
9: **end if**
10: **if** $n'_i \neq 0$ **then**
11: Προσέθεσε την m_i στο $\mathcal{M}'$
12: **end if**
13: **end for**
14: **until** δεν έχει αλλάξει ο βαθμός κάποιας αντιστοίχισης
15: **return** $\mathcal{M}'$

κάτι τέτοιο είναι απαραίτητο κάθε φορά που προστίθεται ένας ασαφής ισχυρισμός, να γίνεται έλεγχος ασυνέπειας. Στην περίπτωση που προκύπτει κάποια ασυνέπεια, είναι απαραίτητο ο βαθμός της αντιστοίχισης που δημιούργησε τον ασαφή ισχυρισμό να τροποποιείται κατάλληλα , έτσι ώστε αυτή να επιλύεται.

7.3.1 Αλγόριθμος Επικύρωσης Αντιστοιχίσεων

Η προσέγγιση μας για την επικύρωση αντιστοιχίσεων ακολουθεί την παρακάτω διαδικασία. Αρχικά ανακτούμε τις αντιστοιχίσεις που παράγονται από κάποιο σύστημα αντιστοίχισης οντολογιών. Οι αντιστοιχίσεις αυτές μπορεί να προκύψουν με χρήση διάφορων μεθόδων, ανάλογα με το σύστημα που χρησιμοποιούμε. Τέτοιες μέθοδοι μπορεί να είναι συντακτικές, δομικές ή ακόμα και σημασιολογικές. Στη συνέχεια, οι αντιστοιχίσεις που έχουν προκύψει ερμηνεύονται ως ασαφείς αναθέσεις όπως παρουσιάστηκε στον Ορισμό 7.2.3 και η οντολογία που προκύπτει με τον εμπλουτισμό της οντολογίας στόχου της αντιστοίχισης με τις αναθέσεις αυτές υπόκειται σε έλεγχο συνέπειας με χρήση ενός συστήματος ασαφούς συλλογιστικής. Τέλος, οι αντιστοιχίσεις επανεξετάζονται με βάση τα αποτελέσματα του συστήματος ασαφούς συλλογιστικής. Πιο συγκεκριμένα ανανεώνονται τα βάρη των αντιστοιχίσεων που προκαλούν ασυνέπειες, ενώ οι υπόλοιπες διατηρούνται ως έχουν.

Ο αλγόριθμος μας φαίνεται στον Αλγόριθμο 15. Δέχεται ως είσοδο ένα σύνολο αντιστοιχίσεων M μαζί με τις οντολογίες $O_1 = \langle \mathcal{T}_1, \mathcal{A}_1 \rangle$ και $O_2 = \langle \mathcal{T}_2, \mathcal{A}_2 \rangle$ τις οποίες αφορά και κατασκευάζει ένα νέο σύνολο αντιστοιχίσεων M'. Πιο αναλυτικά, ο αλγόριθμος μας αρχικά ταξινομεί το σύνολο M σε φθίνουσα διάταξη (γραμμή 2). Με τον τρόπο αυτό εξετάζονται πρώτα οι αντιστοιχίσεις που έχουν μεγαλύτερο βαθμό,

Αλγόριθμος 16 computeStrength$(m_i, \mathcal{O}_1, \mathcal{O}_2)$

Είσοδος: Μια αντιστοίχιση $m_i := \{C_i, C'_i, n_i, R_i\}$ και δύο οντολογίες $\mathcal{O}_1, \mathcal{O}_2$, τ.ω. $C_i \in \mathcal{O}_1$, $C'_i \in \mathcal{O}_2$

1: $\mathcal{O}' = \langle \mathcal{T}_2, \mathcal{A}_2 \rangle$
2: **for all** a τ.ω. $\mathcal{O}_1 \models C_i(a)$ **do**
3: Προσέθεσε την $a : C \bowtie n_i$ στην $\mathcal{O}'$
4: Έλεγξε την $\mathcal{O}'$ για συνέπεια
5: **if** $\mathcal{O}_2$ ασυνεπής **then**
6: Αφαίρεσε όλα τα $a : C \bowtie n_i$ από την $\mathcal{O}'$
7: $s :=$ refineStrength(inconsInfo)
8: **else**
9: $s := n_i$
10: **end if**
11: **end for**
12: **return** s

δηλαδή, οι αντιστοιχίσεις στις οποίες η ομοιότητα είναι μεγαλύτερη. Στη συνέχεια, ο αλγόριθμος εξετάζει κάθε αντιστοίχιση και υπολογίζει ένα νέο βαθμό χρησιμοποιώντας τη συνάρτηση computeStrength (γραμμή 5). Εάν ο βαθμός αυτός είναι διαφορετικός από τον αρχικό (γραμμή 6) τότε η αντιστοίχιση ανανεώνεται και η ίδια διαδικασία πραγματοποιείται εξαρχής. Αυτό γίνεται επειδή η νέα τιμή της αντιστοίχισης μπορεί να προκαλέσει μια νέα ασυνέπεια, που δεν προέκυπτε προηγουμένως. Στην περίπτωση που ο βαθμός που προκύπτει είναι διάφορος του μηδενός τότε η συγκεκριμένη αντιστοίχιση προστίθεται στο τελικό αποτελέσμα (γραμμή 11), διαφορετικά διαγράφεται. Η διαδικασία αυτή επαναλαμβάνεται επαναληπτικά για όλες τις αντιστοιχίσεις μέχρις ότου να εξεταστούν όλες οι αντιστοιχίσεις και να μην προκύπτουν άλλες ασυνέπειες.

Η συνάρτηση computeStrength που χρησιμοποιείται για τον υπολογισμό των νέων βαθμών περιγράφεται από τον Αλγόριθμο 16. Αρχικά κατασκευάζει μια νέα οντολογία $O' = \langle \mathcal{T}', \mathcal{A}' \rangle$, όπου $\mathcal{T}' = \mathcal{T}_2$. Το ABox της νέας οντολογίας κατασκευάζεται σταδιακά από το ABox της O_2 και χρησιμοποιώντας τις υπάρχουσες αντιστοιχίσεις κατασκευάζει αναθέσεις από τα άτομα της οντολογίας O_1, όπως περιγράφεται στον Ορισμό 7.2.4. Για να πραγματοποιηθεί αυτή η διαδικασία είναι απαραίτητη η χρήση ενός συστήματος συμπερασματολογίας για κλασικές ΠΛ όπως είναι το Pellet [146], το FaCT++ [162], το Racer [71], HermiT [20, 59, 106] και άλλα.

Στη συνέχεια, αφού προστεθεί μια νέα ασαφής ανάθεση στην $\mathcal{O}'$ καλείται το σύστημα FiRE [151, 144, 142, 163, 145] έτσι ώστε να γίνει έλεγχος συνέπειας.. Το FiRE είναι ένα πρωτότυπο σύστημα γραμμένο σε JAVA που υλοποιεί έναν ασαφή αλγόριθμο για τη γλώσσα f_{KD}-$\mathcal{SHIN}$ [154] και υποστηρίζει υπηρεσίες συλλογιστικής για έλεγχο συνέπειας μιας ασαφούς βάσης γνώσης, έλεγχο συνεπαγωγής ασαφών αναθέσεων, και έλεγχο υπαγωγής ανάμεσα σε ασαφείς έννοιες. Συνεχίζοντας, λοιπόν, στον αλγόριθμο μας εάν προκύψει κάποια ασυνέπεια ο βαθμός της αντιστοίχισης τροποποιείται. Η διαδικασία που τροποποιεί το βαθμό της αντιστοίχισης είναι η refineStrength, και δέχεται ως είσοδο πληροφορίες από την μηχανή ασαφούς συλλογι-

Αλγόριθμος 17 QueryAnsweringOverMappings($\mathcal{Q}, \mathcal{M}, \mathcal{O}_1, \mathcal{O}_2$)

Είσοδος: Ένα ερώτημα $\mathcal{Q}$ σε μια οντολογία $\mathcal{O}_1$ και ένα σύνολο αντιστοιχίσεων $\mathcal{M}$ της $\mathcal{O}_1$ σε μια άλλη οντολογία $\mathcal{O}_2$

1: $\mathcal{M}' =$ fuzzyValidation($\mathcal{M}, \mathcal{O}_1, \mathcal{O}_2$)
2: **for all** $m_i = \{C_i, C'_i, n_i, R_i\} \in \mathcal{M}'$ **do**
3: **for all** a τ.ω. $\mathcal{O}_1 \models C_i(a)$ **do**
4: Προσέθεσε την $a : C'_i \bowtie n_i$ στην $\mathcal{O}_2$
5: **end for**
6: **end for**
7: **return** cert($\mathcal{Q}, \mathcal{O}_2$)

στικής (FiRE) που αφορούν τις συνθήκες που προκαλούν την ασυνέπεια και με βάση την πληροφορία αυτή προχωρά στην τροποποίηση του βαθμού της αντιστοίχισης έτσι ώστε να αποκατασταθεί η συνέπεια της οντολογίας.

Παράδειγμα 7.3.2. Συνεχίζοντας το Παράδειγμα 7.3.1 ο Αλγόριθμος 15 όταν προστίθενται οι ισχυρισμοί (ΗλεκτρονικήΣυσκευή(κσ$_1$), 0.7) και (ΗλεκτρονικήΣυσκευή(κτ$_1$), 0.7) και λόγω της ασυνέπειας που προκύπτει από τον τελευταίο τροποποιεί το βαθμό της αντιστοίχισης 1 σε 1.0. Συνεχίζοντας, τροποποιείται ο βαθμός της αντιστοίχισης 3, που όπως είδαμε προσθέτει τον ασαφή ισχυρισμό (ΣταθΤηλέφωνο(κτ$_1$), 0.4) προκαλώντας ασυνέπεια. Ο βαθμός της αντιστοίχισης αυτής γίνεται ίσος με 0 με αποτέλεσμα η αντιστοίχιση αυτή να διαγράφεται. Έτσι, τελικά το σύνολο αντιστοιχίσεων $\mathcal{M}$ μετατρέπεται στο σύνολο $\mathcal{M}'$ που αποτελείται από τις παρακάτω αντιστοιχίσεις:

1. map(ΚινητήΣυσκευή, ΗλεκτρονικήΣυσκευή, 1.0)
2. map(ΚινητόΤηλέφωνο, Τηλέφωνο, 1.0)
3. map(ΚινητόΤηλέφωνο, ΚινΤηλέφωνο, 1.0) ♢

7.3.2 Αλγόριθμος Απάντησης Ερωτημάτων μέσω Αντιστοιχίσεων

Έχοντας λύσει το πρόβλημα της ασυνέπειας που προκύπτει από την ασαφή ερμηνεία των αντιστοιχίσεων, είμαστε σε θέση να παρουσιάσουμε τον αλγόριθμο μας για την απάντηση ερωτημάτων μέσω αντιστοιχίσεων. Στο σημείο αυτό τονίζουμε πως ο αλγόριθμος μας δεν αφορά τη διαδικασία επανεγγραφής για την f-DL-Lite που έχει περιγραφεί από τον Straccia [158] ούτε για την απάντηση ερωτημάτων [158, 111] αλλά κυρίως στην επικύρωση των αντιστοιχίσεων και στην κατασκευή της νέας οντολογίας που περιέχει τις ασαφείς αναθέσεις που έχουν προκύψει από τις αντιστοιχίσεις.

Έτσι λοιπόν, ο αλγόριθμος μας, που φαίνεται στον Αλγόριθμο 17, δέχεται ως είσοδο ένα ΣΕ $\mathcal{Q}$, μια αντιστοίχιση $\mathcal{M}$ για δύο οντολογίες $\mathcal{O}_1$ και $\mathcal{O}_2$, αλλά και τις οντολογίες αυτές και παράγει ένα σύνολο απαντήσεων για το ερώτημα $\mathcal{Q}$ με βάση την οντολογία που προκύπτει από την μετάφραση των αντιστοιχίσεων με χρήση της Ασαφούς Συνολοθεωρίας και της Ασαφούς Λογικής. Αρχικα, ο αλγόριθμος μας

εκτελεί τον Αλγόριθμο 15 έτσι ώστε να επικυρώσει το σύνολο αντιστοιχίσεων $\mathcal{M}$ (γραμμή 1). Στη συνέχεια, και εφόσον από τον Αλγόριθμο 15 έχει επιστραφεί ένα νέο σύνολο αντιστοιχίσεων $\mathcal{M}'$ προσθέτει στον οντολογία $\mathcal{O}_2$ όλους τους ασαφείς ισχυρισμούς που προκύπτουν λόγω των αντιστοιχίσεων (γραμμές 2–6). Τέλος, εφόσον η $\mathcal{O}_2$ έχει εμπλουτιστεί γίνεται η αποτίμηση του ερωτήματος $\mathcal{Q}$ και επιστρέφονται οι απαντήσεις.

Παράδειγμα 7.3.3. Συνεχίζοντας το Παράδειγμα 7.3.1 ο αλγόριθμος μας θα επιστρέψει, εκτός από τις απαντήσεις που προκύπτουν από την οντολογία $\mathcal{O}_2$ και τα άτομα $\mathsf{κτ}_1$ και $\mathsf{κσ}_1$ με βαθμό 1.0. $\diamondsuit$

Μέρος V

Επίλογος

Κεφάλαιο 8

Σχετική Βιβλιογραφία

Όσον αφορά στην σχετική με τα προβλήματα που παρουσιάσαμε βιβλιογραφία, εξ όσων γνωρίζουμε δεν υπάρχει αντίστοιχη δουλειά όσον αφορά στην κατασκευή επανεγγραφής για ερωτήματα που έχουν τροποποιηθεί στην παρουσία λογικών περιορισμών τόσο στην περιοχή των οντολογιών όσο και στην περιοχή των βάσεων δεδομένων. Το μόνο πρόβλημα που είναι σχετικό είναι η επανεγγραφή ερωτημάτων για οντολογίες που έχουν τροποποιηθεί με προσθήκη ή αφαίρεση αξιωμάτων [161]. Επιπλέον, ένα άλλο σχετικό πρόβλημα είναι αυτό της *προσαρμογής όψεων* (*view adaptation*) [69, 101], όπου το πρόβλημα είναι ο υπολογισμός της *υλοποίησης των όψεων* (*materialisation of views*) [39, 70] για όψεις οι οποίες έχουν επανακαθοριστεί. Παρόλα αυτά, και στις δύο αυτές δημοσιεύσεις η έμφαση δίνεται στην ανανέωση των δεδομένων και επιπλέον δεν αφορούν στην ύπαρξη λογικών περιορισμών.

Επιπρόσθετα, σε ένα διαφορετικό σενάριο στο Σημασιολογικό Ιστό, έχουν κατασκευαστεί διάφοροι επαναληπτικοί και επαυξητικοί αλγόριθμοι οι οποίοι βοηθούν τους χρήστες να κατασκευάσουν τα ερωτήματα τους [169, 160, 45]. Οι αλγόριθμοι αυτοί υπολογίζουν πιθανά σημασιολογικά ερωτήματα που μπορεί να ρωτήσει ο χρήστης με βάση το ερώτημα που έχει ήδη κάνει και στη συνέχεια του προτείνουν αυτά που θεωρούν επικρατέστερα.

Σχετικά με το πρόβλημα της επανεγγραφής σταθερών, προκαθορισμένων, ερωτημάτων όπως αναφέραμε και στην ενότητα 1 τα τελευταία χρόνια η μελέτη της επανεγγραψιμότητας έχει παρουσιάσει ιδιαίτερη άνθιση τόσο από άποψη μελέτης πολυπλοκότητας [33, 67, 85, 109] όσο και στην κατασκευή συστημάτων επανεγγραφής ερωτημάτων [31, 116, 132, 40, 64, 108, 49, 127]. Στη συνέχεια παρουσιάζουμε μια επισκόπηση των υπαρχόντων συστημάτων επανεγγραφής. Μια πιο αναλυτική έρευνα στην απάντηση ερωτημάτων μπορεί να βρεθεί στο [65].

Ο πρώτος αλγόριθμος επανεγγραφής συζευκτικών ερωτημάτων κατασκευάστηκε από τους Calvanese et al. [32, 31] για την οικογένεια γλωσσών της DL-Lite που υλοποιήθηκε στη συνέχεια στο σύστημα Quonto [2]. Ο αλγόριθμος αυτός, που ονομάζεται PerfectRef, διαχωρίζει το TBox από το ABox. Το ερώτημα επανεγγράφεται με βάση τα αξιώματα του TBox σε ένα σύνολο από ερωτήματα στη συνέχεια που αποτιμούνται με βάση το ABox, το οποίο μπορεί να είναι αποθηκευμένο σε μια σχεσιακή βάση δεδομένων. Με τη μέθοδο αυτή η απάντηση ερωτημάτων ανάγεται στην αποτίμηση ερωτημάτων πάνω σε ένα στιγμιότυπο μιας βάσης δεδομένων χωρίς να είναι

απαραίτητη η ύπαρξη του TBox μιας οντολογίας. Για το λόγο αυτό η πολυπλοκότητα του αλγορίθμου ως προς τα δεδομένα είναι ίδια με την πολυπλοκότητα της αποτίμησης του νέου ερωτήματος. Πιο συγκεκριμένα ο αλγόριθμος PerfectRef επανεγγράφει ένα ερώτημα σε ένα σύνολο ερωτημάτων χρησιμοποιώντας ένα *βήμα αναδιατύπωσης* (*reformulation step*) και ένα *βήμα μείωσης* (*reduction step*). Με την εργασία τους αυτή οι Calvanese et al. άνοιξαν το δρόμο για τη χρήση των τεχνολογιών αποθήκευσης, δεικτοδότησης και αποτίμησης δεδομένων που χρησιμοποιούνται εδώ και αρκετά χρόνια από τις βάσεις δεδομένων στις τεχνολογίες του Σημασιολογικού Ιστού.

Στη συνέχεια ο Rosati [129] πρότεινε ένα αλγόριθμο επανεγγραφής συζευκτικών ερωτημάτων για την Περιγραφική Λογική $\mathcal{EL}$. Με τον αλγόριθμο αυτό ο Rosati έδειξε ότι η επανεγγραφή συζευκτικών ερωτημάτων για την $\mathcal{EL}$ ανήκει στην κλάση πολυπλοκότητας PTIME-complete ως προς την πολυπλοκότητα των δεδομένων [129, 88, 90]. Ο αλγόριθμος αυτός μετατρέπει ένα συζευκτικό ερώτημα και ένα TBox σε ένα πρόγραμμα Datalog. Αυτό έχει σαν αποτέλεσμα για την αποτίμηση του προγράμματος Datalog να είναι απαραίτητη μια επαγωγική βάση δεδομένων.

Συνεχίζοντας οι Pérez-Urbina et al. [117, 116] παρουσίασαν διάφορους αλγορίθμους για επανεγγραφή ερωτημάτων στις γλώσσες DL-Lite, DL-Lite$^+$, όπου από ένα συζευκτικό ερώτημα παράγεται ένα σύνολο από συζευκτικά ερωτήματα καθώς και στην $\mathcal{ELHI}$ όπου από ένα συζευκτικό ερώτημα παράγεται ένα πρόγραμμα Datalog. Οι αλγόριθμοι αυτοί, που στηρίζουν τη λειτουργία τους στη μέθοδο της ανάλυσης πρώτης-τάξης, υλοποιήθηκαν στο σύστημα Requiem [116] και οι πειραματικές μελέτες έδειξαν ότι αποδίδουν καλύτερα από το PerfectRef. Το Requiem, τέλος, είναι το πρώτο σύστημα που εισήγαγε κριτήρια περιττότητας (redundancy criteria), όπως είναι η αφαίρεση ερωτημάτων που μειώνει το μέγεθος των συζευκτικών ερωτημάτων που παράγονται αποβάλλοντας αυτά που είναι περιττά [126].

Παρόλα αυτά οι τεχνικές αυτές παρουσιάζουν ένα σημαντικό μειονέκτημα. Το μέγεθος της επανεγγραφής που υπολογίζεται αυξάνεται εκθετικά σε σχέση με τον αριθμό των ατόμων του αρχικού συζευκτικού ερωτήματος. Για το λόγο αυτό οι Rosati και Almatelli [132] παρουσίασαν το Presto που αποτελεί ένα βελτιστοποιημένο σύστημα για επανεγγραφή συζευκτικών ερωτημάτων για DL-Lite Περιγραφικές Λογικές. Το Presto αποφεύγει την εκθετική αύξηση του μεγέθους της επανεγγραφής αφού αντί για ένα σύνολο από συζευκτικά ερωτήματα παράγει ένα μη-αναδρομικό πρόγραμμα Datalog μεταφέροντας με αυτό τον τρόπο μέρος της υπολογιστικής διαδικασίας στο σύστημα της βάσης δεδομένων.

Επιπλέον, οι Stamou et al. [149] εισήγαγαν μια μέθοδο η οποία ακολουθεί μια διαφορετική, προοδευτική, προσέγγιση για την επανεγγραφή ερωτημάτων. Σύμφωνα με τη μέθοδο αυτή από το αρχικό ερώτημα πρώτα παράγονται τα ερωτήματα τις επανεγγραφής που είναι πιο «σχετικά» με αυτό και στη συνέχεια παράγονται ερωτήματα που είναι λιγότερο «σχετικά».

Πρόσφατα, οι Chortaras et al. [41, 40] παρουσίασαν ένα βελτιστοποιημένο αλγόριθμο επανεγγραφής για την παραγωγή ενός συνόλου συζευκτικών ερωτημάτων από ένα συζευκτικό ερώτημα και μια οντολογία DL-Lite που ονομάζεται Rapid. Η αποδοτικότητα του αλγορίθμου αυτού οφείλεται στην επιλεκτική εφαρμογή του κανόνα της ανάλυσης, που αντί να εφαρμόζεται αδιακρίτως, χρησιμοποιείται μόνο σε περι-

πτώσεις που πρόκειται να παραχθούν χρήσιμα, μη-περιττά συζευκτικά ερωτήματα. Με τον τρόπο αυτό αποφεύγεται η παραγωγή ενός πολύ μεγάλου αριθμού συζευκτικών ερωτημάτων εφαρμόζοντας τελικά τους ελέγχους υπαγωγής, που αποτελούν ιδιαίτερα ακριβή διαδικασία, σε μικρότερο αριθμό συζευκτικών ερωτημάτων. Οι πειραματικές μετρήσεις έδειξαν ότι το Rapid υπερτερεί αισθητά από τα υπόλοιπα συστήματα [41, 40].

Ακολούθως, οι Rodriquez-Muro et al. [127] παρουσίασαν μια τεχνική που έχει ως σκοπό την επίτευξη πληρότητας για το ABox επιτρέποντας ταυτόχρονα την δημιουργία αποδοτικών ερωτημάτων SQL χωρίς η διαδικασία αυτή να είναι δαπανηρή. Στη συνέχεια, παρουσίασαν μια προσέγγιση με σύμφωνα με την οποία λαμβάνουν υπ' όψιν την πληρότητα των δεδομένων σε σχέση με το DL-Lite TBox. Χρησιμοποιώντας την προσέγγιση αυτή τα TBox που παράγονται είναι βέλτιστα, με την έννοια ότι περιέχουν μόνο τις αναθέσεις που είναι απαραίτητες για να διασφαλιστεί η πληρότητα των υπολογιζόμενων απαντήσεων. Επιπλέον, παρουσιάζουν ένα μηχανισμό απολοποίησης του λεξιλογίου της οντολογίας που μειώνει την πολυπλοκότητα των διεργασιών συλλογιστικής με βάση ισοδύναμες έννοιες και ρόλους. Χρησιμοποιώντας αυτές τις τεχνικές, οι Rodriquez-Muro et al. κατασκεύασαν το σύστημα επανεγγραφής ερωτημάτων Quest.

Τέλος, οι Eiter et al. [49] παρουσίασαν το σύστημα επανεγγραφής Clipper που υποστηρίζει την επανεγγραφή ερωτημάτων για οντολογίες Horn-$\mathcal{SHIQ}$ [91, 110]. Ο αλγόριθμος τους, εφαρμόζει αρχικά έναν ειδικό λογισμό ανάλυσης και στη συνέχεια επανεγγράφει τα ερωτήματα με βάση το TBox που έχει αναλυθεί σε ένα πρόγραμμα Datalog το οποίο μπορεί να αποτιμηθεί σε ένα οποιοδήποτε ABox. Επιπλέον, ο αλγόριθμος αυτός υποστηρίζει μεταβατικούς ρόλους, οι οποίοι εμφανίζονται αρκετά συχνά σε πραγματικές εφαρμογές [133], ενώ στην απάντηση ερωτημάτων η χρήση τους περιέχει ένα βαθμό δυσκολίας [60, 48].

Πέρα από την απάντηση συζεκτικών ερωτημάτων για Περιγραφικές Λογικές ένας άλλος τομέας που έχει παρουσιάσει άνθιση τα τελευταία χρόνια είναι η επανεγγραφή ερωτημάτων πάνω με χρήση περιορισμών βάσεων δεδομένων. Οι Calí et al. [25, 28] μελέτησαν και παρουσίασαν αλγόριθμους επανεγγραφής ερωτημάτων πάνω από διάφορες κλάσσεις *εξαρτήσεων παραγωγής πλειάδων* (*tuple generating dependencies*) αλλά και με χρήση περιγραφών που βασίζονται στο μοντέλο *Οντοτήτων Συσχετίσεων* (*Entity Relationship model*) [27, 29]. Επιπρόσθετα οι Gottlob et al. παρουσίασαν επίσης έναν πρακτικό αλγόριθμο για επανεγγραφή ερωτημάτων για τη γραμμική-Datalog$^{\pm}$ που υλοποιείται στο σύστημα Nyaya [64]. Ο αλγόριθμος αυτός στηρίζει τη λειτουργία του στον αρχικό αλγόριθμο PerfectRef για την DL-Lite αλλά τον βελτιώνει κάνοντας χρήση ποικίλων βελτιστοποιήσεων, όπως η παραγοντοποίηση ατόμων (atom factorization) με την οποία μειώνεται το μέγεθος των περιττών ερωτημάτων κατά το βήμα μείωσης.

Τέλος, μια διαφορετική προσέγγιση, που ονομάζεται *συνδυαστική επανεγγραφή* (*combined rewriting*), προτάθηκε από τους Lutz et al. [94] για τη γλώσσα $\mathcal{EL}$ και από τους Kontchakov et al. [87] για τη γλώσσα DL-Lite$^{\mathcal{N}}_{horn}$. Ο αλγόριθμος αυτός παράγει μικρά σύνολα επανεγγραφής και μπορεί να χρησιμοποιηθεί επίσης και σε γλώσσες που δεν είναι επανεγγράψιμες σε πρώτη-τάξη, αλλά απαιτεί μια προεπεξεργασία της βάσης δεδομένων.

Όσον αφορά στο πρόβλημα της επικύρωσης αντιστοιχίσεων υπάρχουν διάφορες μελέτες που έχουν εστιάσει στην επικύρωση αντιστοιχίσεων σε σχέση με τη συμασιολογία των εμπλεκόμενων οντολογιών διατηρώντας ταυτόχρονα την αβέβαιη φύση των αντιστοιχίσεων. Στην [30] παρουσιάζεται μια γλώσσα για την αναπαράσταση και την συλλογιστική με αβέβαιες αντιστοιχίσεις συνδυάζοντας τις οντολογίες με γλώσσες κανόνων και με πιθανοτική συλλογιστική. Η μέθοδος αυτή αναπαριαστά τιμές εμπιστοσύνης των αντιστοιχίσεων ως πιθανότητες λάθους έτσι ώστε να επιλυθούν οι ασυνέπειες χρησιμοποιώντας πιθανότητες εμπιστοσύνης. Επιπλέον, σε μια άλλη μελέτη παρουσιάστηκε ένα εργαλείο για την επικύρωση αντιστοιχίσεων με τη χρήση πιθανοτικής συλλογιστικής [36]. Η ιδέα πίσω από αυτό ήταν η ερμηνεία των αντιστοιχίσεων ως πιθανοτικές και υποθετικές σχέσεις ανάμεσα στα στοιχεία των οντολογιών έτσι ώστε να κατασκευαστεί μια οντολογία από τις δύο ανεξάρτητες οντολογίες και στη συνέχεια να γίνει έλεγχος για ασυνέπειες.

Κεφάλαιο 9

Συνεισφορά και Θέματα προς Έρευνα

Η μελέτη της απάντησης Συζευκτικών Ερωτημάτων σε πολύ μεγάλα ABox, που πραγματοποιείται μέσω της επανεγγραφής ερωτημάτων, έχει παρουσιάσει μεγάλη άνθιση τα τελευταιά χρόνια με αποτέλεσμα τη δημιουργία πολλών και αποδοτικών αλγορίθμων [31, 116, 132, 40, 64, 108, 49, 127]. Επιπλέον ιδιαίτερα συχνό είναι το φαινόμενο κατά το οποίο οι χρήστες για να βρουν αυτό που ζητάνε θέτουν αρχικά ένα «γενικό» ερώτημα και στη συνέχεια το τροποποιούν κατάλληλα ανάλογα με τα αποτελέσματα που παίρνουν. Συμπεραίνουμε λοιπόν ότι το τελικό ερώτημα δεν είναι πάντα γνωστό εκ των προτέρων. Αυτό το γεγονός έχει ως αποτέλεσμα τα διάφορα συστήματα/αλγόριθμοι που έχουν κατασκευαστεί να εκτελούνται εξαρχής κάθε φορά, που ο χρήστης θέτει ένα νέο τροποποποιημένο ερώτημα, αγνοώντας τις πληροφορίες που έχουν ήδη παραχθεί. Ο κύριος στόχος της εργασίας μας ήταν η μελέτη και ανάπτυξη αποδοτικών αλγορίθμων που επιτρέπουν την επανεγγραφή ερωτημάτων που έχουν τροποποιηθεί χρησιμοποιώντας πληροφορία που έχει ήδη παραχθεί προηγουμένως.

Στην διατριβή αυτή συνεισφέρουμε σημαντικά προς την κατεύθυνση της αποδοτικής επανεγγραφής τροποποιημένων ερωτημάτων. Πιο συγκεκριμένα, παρουσιάσαμε αλγορίθμους που ισχύουν για όλες τις γλώσσες που είναι επανεγγράψιμες, οι οποίοι διαχειρίζονται τα προβλήματα της κατασκευής μιας επανεγγραφής για ερωτήματα στα οποία έχουν προστεθεί/αφαιρεθεί διακεκριμένες μεταβλητές αλλά και έχουν προστεθεί/αφαιρεθεί άτομα.

Η συνεισφορά της εργασίας μας συνοψίζεται παρακάτω:

- *Μείωση ερωτημάτων με διακεκριμένες μεταβλητές:* Μελετήσαμε το πρόβλημα υπολογισμού μιας επανεγγραφής για ερωτήματα που έχουν μειωθεί με διακεκριμένες μεταβλητές. Πιο συγκεκριμένα, δεδομένης μιας επανεγγράψιμης οντολογίας, ενός συζευκτικού ερωτήματος, μιας ήδη υπολογισμένης επανεγγραφής για το ερώτημα αυτό και ενός συνόλου διακεκριμένων μεταβλητών μελετήσαμε πώς μπορούμε να υπολογίσουμε μια επανεγγραφή για το νέο ερώτημα χρησιμοποιώντας την επανεγγραφή που έχουμε σαν είσοδο αποφεύγοντας να την υπολογίσουμε εξαρχής. Μελετήσαμε το πρόβλημα θεωρητικά και στη συνέχεια

σχεδιάσαμε ένα αποδοτικό αλγόριθμο. Σε γενικές γραμμές, ο αλγόριθμος μας εφαρμόζει τον τελεστή της προβολής στα ερωτήματα της αρχικής επανεγγραφής και στη συνέχεια στην περίπτωση που από τα ερωτήματα που προκύπτουν «ενεργοποιούνται» νέοι κανόνες συμπερασμού εφαρμόζει, επιλεκτικά, έναν αλγόριθμο επανεγγραφής, έτσι ώστε να παράγει τα απαραίτητα ερωτήματα.

- *Επέκταση ερωτημάτων με διακεκριμένες μεταβλητές:* Μελετήσαμε το πρόβλημα υπολογισμού μιας επανεγγραφής για ερωτήματα που έχουν επεκταθεί με διακεκριμένες μεταβλητές. Πιο συγκεκριμένα, δεδομένης μιας επανεγγράψιμης οντολογίας, ενός συζευκτικού ερωτήματος, μιας ήδη υπολογισμένης επανεγγραφής για το ερώτημα αυτό και ενός συνόλου μεταβλητών μελετήσαμε πώς μπορούμε να υπολογίσουμε μια επανεγγραφή για το νέο ερώτημα εκμεταλλευόμενοι την επανεγγραφή που έχουμε σαν είσοδο αποφεύγοντας να την υπολογίσουμε εξαρχής. Μελετήσαμε το πρόβλημα θεωρητικά και στη συνέχεια σχεδιάσαμε ένα αποδοτικό αλγόριθμο. Σε γενικές γραμμές, ο αλγόριθμος μας επεκτείνει με τις νέες διακεκριμένες μεταβλητές όλα τα ερωτήματα της αρχικής επανεγγραφής και κρατάει στην τελική επανεγγραφή μόνο αυτά τα οποία είναι ασφαλή.

- *Μείωση ερωτημάτων με αφαίρεση ατόμου:* Μελετήσαμε το πρόβλημα υπολογισμού μιας επανεγγραφής για ερωτήματα που έχουν μειωθεί με άτομα. Πιο συγκεκριμένα, δεδομένης μιας επανεγγράψιμης οντολογίας, ενός συζευκτικού ερωτήματος, μιας ήδη υπολογισμένης επανεγγραφής για το ερώτημα αυτό και ενός συνόλου από άτομα που υπάρχουν στο σώμα του ερωτήματος μελετήσαμε πώς μπορούμε να υπολογίσουμε μια επανεγγραφή για το νέο ερώτημα εκμεταλλευόμενοι την επανεγγραφή που έχουμε σαν είσοδο αποφεύγοντας να την υπολογίσουμε εξαρχής. Και πάλι, μελετήσαμε το πρόβλημα θεωρητικά και στη συνέχεια σχεδιάσαμε ένα αποδοτικό αλγόριθμο. Ο αλγόριθμος μας, σε γενικές γραμμές, ελέγχει αρχικά εάν η παραγωγή κάποιου ερωτήματος της αρχικής επανεγγραφής είναι ανεξάρτητη από την ύπαρξη του ατόμου που πρόκειται να αφαιρεθεί. Στην περίπτωση είναι ανεξάρτητη τότε κρατάει το συγκεκριμένο ερώτημα. Στην περίπτωση που δεν είναι τότε ελέγχει εάν το ερώτημα έχει προκύψει με συμμετοχή του ατόμου αυτού και στην περίπτωση αυτή το απορρίπτει από την τελική επανεγγραφή. Τέλος εάν το ερώτημα έχει προκύψει χωρίς συμμετοχή του ατόμου, αφαιρεί το άτομο από το συγκεκριμένο ερώτημα και προσθέτει το αποτέλεσμα στην τελική επανεγγραφή. Για τη σωστή λειτουργία και αυτού του αλγόριθμου είναι απαραίτητη η επιλεκτική χρήση ενός εξωτερικού αλγόριθμου επανεγγραφής.

- *Επέκταση ερωτημάτων με προσθήκη ατόμου:* Μελετήσαμε το πρόβλημα υπολογισμού μιας επανεγγραφής για ερωτήματα που έχουν επεκταθεί με άτομα. Πιο συγκεκριμένα, δεδομένης μιας επανεγγράψιμης οντολογίας, ενός συζευκτικού ερωτήματος, μιας ήδη υπολογισμένης επανεγγραφής για το ερώτημα αυτό και ενός συνόλου από άτομα που υπάρχουν μελετήσαμε πώς μπορούμε να υπολογίσουμε μια επανεγγραφή για το νέο ερώτημα χρησιμοποιώντας την επανεγγραφή που έχουμε σαν είσοδο αποφεύγοντας να την υπολογίσουμε εξαρχής. Και πάλι,

μελετήσαμε το πρόβλημα θεωρητικά και στη συνέχεια σχεδιάσαμε ένα αποδοτικό αλγόριθμο. Ο αλγόριθμος μας, σε γενικές γραμμές, προσθέτει το άτομο αυτό στα ερωτήματα της αρχικής επανεγγραφής και σε συγκεκριμένες περιπτώσεις, όπου «ενεργοποιούνται» νέοι κανόνες συμπερασμού, εκτελεί και αυτό κάποιον αλγόριθμο επανεγγραφής.

- *Τεχνικές βελτιστοποίησης αλγορίθμων επανεγγραφής τροποποιημένων ερωτημάτων:* Στη συνέχεια, έτσι ώστε να βελτιώσουμε τη συμπεριφορά των παραπάνω αλγορίθμων παρουσιάσαμε ένα σύνολο από βελτιστοποιήσεις οι οποίες καθιστούν τους αλγορίθμους μας ιδιαίτερα αποδοτικούς.

- *Βελτιστοποίηση ερωτημάτων με προσθήκη ατόμου:* Συνεχίζοντας, και εστιάζοντας στο πρόβλημα της επέκτασης ερωτημάτων με άτομα, μελετήσαμε το πρόβλημα αυτό για την περιορισμένης εκφραστικότητας γλώσσσα DL-Lite$_R$. Πιο συγκεκριμένα, δεδομένης μιας οντολογίας DL-Lite$_R$, ενός συζευκτικού ερωτήματος, μιας ήδη υπολογισμένης επανεγγραφής για το ερώτημα αυτό και ενός ατόμου μελετήσαμε πώς μπορούμε να υπολογίσουμε μια επανεγγραφή για το νέο ερώτημα επεκτείνοντας την επανεγγραφή που έχουμε σαν είσοδο αποφεύγοντας να την υπολογίσουμε εξαρχής. Μελετήσαμε το πρόβλημα θεωρητικά και στη συνέχεια σχεδιάσαμε ένα αποδοτικό αλγόριθμο. Σε γενικές γραμμές, ο αλγόριθμος υπολογίζει την ένωση συζευκτικών ερωτημάτων για ένα ερώτημα που είναι σχετικό *μόνο* με το νεοεισηχθέν άτομο και στη συνέχεια το συνδυάζει με την ένωση συζευκτικών ερωτημάτων που δίνεται σαν είσοδος χρησιμοποιώντας κατάλληλες πράξεις. Με τον τρόπο αυτό ο αλγόριθμος μας εκτελεί *μόνο* την πρόσθετη εργασία που είναι απαραίτητη για τον υπολογισμό μιας επανεγγραφής για το επεκτεταμένο ερώτημα.

- *Κατασκευή αλγορίθμου επανεγγραφής ερωτημάτων:* Οι τεχνικές μας για την επανεγγραφή επεκτατμένων ερωτημάτων υποδηλώνουν ότι είναι εφικτή μια πρωτότυπη προσέγγιση για τον υπολογισμό ενός συνόλου συζευκτικών ερωτημάτων για *προκαθορισμένα* ερωτήματα για οντολογίες DL-Lite$_R$. Πιο συγκεκριμένα, δεδομένου ενός (προκαθορισμένου) ερωτήματος δείξαμε ότι μπορεί κανείς να επιλέξει ένα από τα άτομά του, να υπολογίσει ένα σύνολο συζευκτικών ερωτημάτων για αυτό, και στη συνέχεια να προσθέσει επαναληπτικά τα υπόλοιπα άτομα επεκτείνοντας κάθε φορά την προηγουμένως υπολογισμένη επανεγγραφή. Όταν όλα τα άτομα έχουν επεξεργαστεί θα έχει παραχθεί μια επανεγγραφή για το αρχικό ερώτημα. Βασισμένοι σε αυτή την ιδέα παρουσιάσαμε ένα αναλυτικό αλγόριθμο επανεγγραφής ερωτημάτων για συζευκτικά ερωτήματα σε οντολογίες DL-Lite$_R$.

- *Τεχνικές βελτιστοποίησης αλγόριθμου επανεγγραφής ερωτημάτων:* Στη συνέχεια, έτσι ώστε να βελτιώσουμε την απόδοση του παραπάνω αλγορίθμου παρουσιάσαμε ένα σύνολο από βελτιστοποιήσεις οι οποίες καθιστούν το σύστημα μας ιδιαίτερα γρήγορο και αποδοτικό, μιας και υπερτερεί όλων των ήδη υπάρχοντων συστημάτων.

- *Απάντηση Ερωτήματων μέσω Δικτύου Οντολογιών:* Τέλος, στην διατριβή αυτή μελετήσαμε το πρόβλημα της απάντησης ερωτημάτων πάνω από ένα δίκτυο οντολογιών οι οποίες έχουν αντιστοιχιστεί. Πιο συγκεκριμένα, μελετήσαμε πώς μπορεί να χρησιμοποιηθεί η Ασαφής Συνολοθεωρία και Λογική για την ερμηνεία και επικύρωση των αντιστοιχίσεων ανάμεσα σε δύο οντολογίες ώστε να μπορέσουμε να επιλύσουμε ασυνέπειες που μπορεί να προκύψουν από την αντιστοίχιση αυτή. Στη συνέχεια εφόσον οι ασυνέπειες αυτές έχουν επιλυθεί είναι εφικτή η απάντηση ερωτημάτων πάνω από το δίκτυο των οντολογιών.

Στο υπόλοιπο του κεφαλαίου αυτού παρουσιάζουμε τα θέματα με τα οποία δεν ασχοληθήκαμε στη διατριβή αυτή και τα οποία αποτελούν ανοικτά θέματα για περαιτέρω έρευνα:

- *Επαυξητική επανεγγραφή ερωτημάτων για πιο εκφρασιτκές οντολογίες:* Όπως είδαμε στο κεφάλαιο 5 στη διατριβή αυτή παρουσιάσαμε έναν αλγόριθμο επανεγγραφής σταθερών ερωτημάτων για DL-Lite$_R$ οντολγίες. Παρόλα αυτά η εκφραστικότητα των οντολογιών αυτών είναι σχετικά περιορισμένη και λόγω του ότι τα αποτελέσματα μας για την αποδοτικότητα του αλγορίθμου μας για την DL-Lite$_R$ είναι ιδιαίτερα ενθαρρυντικά παρέχεται η δυνατότητα για μελέτη αλγορίθμων που χρησιμοποιούν την επαυξητική αυτή τεχνική για πιο εκφραστικές οντολογίες όπως $\mathcal{ELHI}$ και Horn-$\mathcal{SHIQ}$ σε Datalog και disjunctive Datalog αντίστοιχα.

- *Απάντηση ερωτημάτων:* Η επανεγγραφή ερωτημάτων αφορά μόνο ένα σκέλος της διαδικασίας απάντησης ερωτημάτων. Για το λόγο αυτό, ιδιαίτερο ενδιαφέρον παρουσιάζει η μελέτη του προβλήματος της απάντησης ερωτημάτων με βάση τον επαυξητικό αλγόριθμο επανεγγραφής που έχουμε κατασκευάσει. Πιο συγκεκριμένα θα ήταν πολύ σημαντικό να μελετηθεί το πώς η στοχευμένη στρατηγική του αλγορίθμου αυτού μπορεί να βοηθήσει στην αποδοτικότερη απάντηση ερωτημάτων για DL-Lite$_R$ οντολογίες.

Παράρτημα Α΄

Ερωτήματα Αξιολόγησης

Οντολογίες DL-Lite

V

$$
\begin{aligned}
\text{Location}(x) &\rightarrow \mathcal{Q}_1(x) \\
\text{Military} - \text{Person}(x) \wedge \text{hasRole}(y,x) \wedge \text{related}(x,z) &\rightarrow \mathcal{Q}_2(x,y) \\
\text{Time} - \text{Dependant} - \text{Relation}(x) \wedge \\
\text{hasRelationMember}(x,y) \wedge \text{Event}(y) &\rightarrow \mathcal{Q}_3(x,y) \\
\text{Object}(x) \wedge \text{hasRole}(x,y) \wedge \text{Symbol}(y) &\rightarrow \mathcal{Q}_4(x,y) \\
\text{Individual}(x) \wedge \text{hasRole}(x,y) \wedge \\
\text{Scientist}(y) \wedge \text{hasRole}(x,z) \wedge \\
\text{Discoverer}(z) \wedge \text{hasRole}(x,k) \wedge \text{Inventor}(k) &\rightarrow \mathcal{Q}_5(x)
\end{aligned}
$$

P1/P5/P5X

$$
\begin{aligned}
\text{edge}(x,y) &\rightarrow \mathcal{Q}_1(x) \\
\text{edge}(x,y) \wedge \text{edge}(y,z) &\rightarrow \mathcal{Q}_2(x) \\
\text{edge}(x,y) \wedge \text{edge}(y,z) \wedge \text{edge}(z,k) &\rightarrow \mathcal{Q}_3(x) \\
\text{edge}(x,y) \wedge \text{edge}(y,z) \wedge \text{edge}(z,k) \wedge \text{edge}(k,l) &\rightarrow \mathcal{Q}_4(x) \\
\text{edge}(x,y) \wedge \text{edge}(y,z) \wedge \text{edge}(z,k) \wedge \text{edge}(k,l) \wedge \text{edge}(l,m) &\rightarrow \mathcal{Q}_5(x)
\end{aligned}
$$

S

$$
\begin{aligned}
\text{StockExchangeMember}(x) &\rightarrow \mathcal{Q}_1(x) \\
\text{Person}(x) \wedge \text{hasStock}(x,y) \wedge \text{Stock}(y) &\rightarrow \mathcal{Q}_2(x) \\
\text{FinantialInstrument}(x) \wedge \text{belongsToCompany}(x,y) \wedge \text{Company}(y) \wedge \\
\text{hasStock}(y,z) \wedge \text{Stock}(z) &\rightarrow \mathcal{Q}_3(x,y,z) \\
\text{Person}(x) \wedge \text{hasStock}(x,y) \wedge \\
\text{Stock}(y) \wedge \text{isListedIn}(y,z) \wedge \text{StockExchangeList}(z) &\rightarrow \mathcal{Q}_4(x,y,z) \\
\text{FinantialInstrument}(x) \wedge \text{belongsToCompany}(x,y) \wedge \\
\text{Company}(y) \wedge \text{hasStock}(y,z) \wedge \text{Stock}(z) \wedge \\
\text{isListedIn}(y,k) \wedge \text{StockExchangeList}(k) &\rightarrow \mathcal{Q}_5(x,y,z,k)
\end{aligned}
$$

U/UX

$$\begin{aligned}
\text{worksFor}(x,y) \wedge \text{affiliatedOrganizationOf}(y,z) &\rightarrow \mathcal{Q}_1(x)\\
\text{Person}(x) \wedge \text{teacherOf}(x,y) \wedge \text{Course}(y) &\rightarrow \mathcal{Q}_2(x,y)\\
\text{Student}(x) \wedge \text{advisor}(x,y) \wedge \text{FacultyStaff}(y) \wedge &\rightarrow \mathcal{Q}_3(x,y,z)\\
\text{takesCourse}(x,z) \wedge \text{teacherOf}(y,z) \wedge \text{Course}(z) &\\
\text{Person}(x) \wedge \text{worksFor}(x,y) \wedge \text{Organization}(y) &\rightarrow \mathcal{Q}_4(x,y)\\
\text{Person}(x) \wedge \text{worksFor}(x,y) \wedge \text{University}(y) \wedge \text{hasAlumnus}(y,x) &\rightarrow \mathcal{Q}_5(x)
\end{aligned}$$

A/AX

$$\begin{aligned}
\text{Device}(x) \wedge \text{assistsWith}(x,y) &\rightarrow \mathcal{Q}_1(x)\\
\text{Device}(x) \wedge \text{assistsWith}(x,y) \wedge \text{UpperLimbMobility}(y) &\rightarrow \mathcal{Q}_2(x)\\
\text{Device}(x) \wedge \text{assistsWith}(x,y) \wedge &\rightarrow \mathcal{Q}_3(x)\\
\text{Hear}(y) \wedge \text{affects}(z,y) \wedge \text{Autism}(z) &\\
\text{Device}(x) \wedge \text{assistsWith}(x,y) \wedge \text{PhysicalAbility}(y) &\rightarrow \mathcal{Q}_4(x)\\
\text{Device}(x) \wedge \text{assistsWith}(x,y) \wedge &\rightarrow \mathcal{Q}_5(x)\\
\text{PhysicalAbility}(y) \wedge \text{affects}(z,y) \wedge \text{Quadriplegia}(z) &
\end{aligned}$$

Οντολογίες $\mathcal{ELHI}$

L

$$\begin{aligned}
\text{GraduateStudent}(x) \wedge \text{takesCourse}(x,y) &\rightarrow \mathcal{Q}_1(x,y)\\
\text{GraduateStudent}(x) \wedge \text{University}(y) \wedge &\rightarrow \mathcal{Q}_2(x,y,z)\\
\text{Department}(z) \wedge \text{memberOf}(x,z) \wedge &\\
\text{subOrganizationOf}(z,y) \wedge \text{undergraduateDegreeFrom}(x,y) &\\
\text{Publication}(x) \wedge \text{publicationAuthor}(x,y) &\rightarrow \mathcal{Q}_3(x,y)\\
\text{Professor}(x) \wedge \text{worksFor}(x,l) \wedge &\rightarrow \mathcal{Q}_4(x,y,z,k,l)\\
\text{name}(x,y) \wedge \text{emailAddress}(x,z) \wedge &\\
\text{telephone}(x,k) &\\
\text{Person}(x) \wedge \text{memberOf}(x,y) &\rightarrow \mathcal{Q}_5(x,y)\\
\text{Student}(x) &\rightarrow \mathcal{Q}_6(x)\\
\text{Student}(x) \wedge \text{Course}(y) \wedge \text{teacherOf}(z,y) \wedge \text{takesCourse}(x,y) &\rightarrow \mathcal{Q}_7(x,y,z)\\
\text{Student}(x) \wedge \text{Department}(y) \wedge \text{memberOf}(x,y) \wedge &\rightarrow \mathcal{Q}_8(x,y,z,k)\\
\text{subOrganizationOf}(y,k) \wedge \text{emailAddress}(x,z) &\\
\text{Student}(x) \wedge \text{Faculty}(y)\text{Course}(z) \wedge &\rightarrow \mathcal{Q}_9(x,y,z)\\
\text{advisor}(x,y) \wedge \text{takesCourse}(x,z) \wedge \text{teacherOf}(y,z) &\\
\text{Student}(x) \wedge \text{takesCourse}(x,y) &\rightarrow \mathcal{Q}_{10}(x,y)\\
\text{ResearchGroup}(x) \wedge \text{subOrganizationOf}(x,y) &\rightarrow \mathcal{Q}_{11}(x,y)\\
\text{Chair}(x) \wedge \text{Department}(y) \wedge &\rightarrow \mathcal{Q}_{12}(x,y,z)\\
\text{worksFor}(x,y) \wedge \text{subOrganizationOf}(y,z) &\\
\text{Person}(x) \wedge \text{hasAlumnus}(y,x) &\rightarrow \mathcal{Q}_{13}(x,y)\\
\text{UndergraduateStudent}(x) &\rightarrow \mathcal{Q}_{14}(x)
\end{aligned}$$

$\mathbf{G}_1$

$$
\begin{aligned}
\text{FunctionalAttribute}(x,y) &\rightarrow \mathcal{Q}_1(x)\\
\text{isCountConcentrationOf}(x,y) \wedge \text{Cell}(y) \wedge &\rightarrow Q_2(x)\\
\text{isMixedThroughout}(y,z) \wedge \text{Tissue}(z) &\\
\text{hasFeature}(x,y) \wedge \text{ProcessFeature}(y) &\rightarrow \mathcal{Q}_3(x)\\
\text{Displacement}(x) \wedge \text{isOutcomeOf}(x,y) \wedge &\rightarrow \mathcal{Q}_4(x)\\
\text{hasDuration}(y,z) \wedge \text{ErythrocyteSedimentation}(y) \wedge &\\
\text{Duration}(z) \wedge \text{hasQuantity}(z,k) \wedge \text{OneHour}(k) &\\
\text{Behaviour}(x) \wedge \text{hasGoal}(x,y) &\rightarrow \mathcal{Q}_5(x)
\end{aligned}
$$

$\mathbf{G}_2$

$$
\begin{aligned}
\text{BodyPart}(x) \wedge \text{hasIntrinsicAbnormalityStatus}(x,y) &\rightarrow \mathcal{Q}_1(x)\\
\text{Artery}(x) \wedge \text{hasFeature}(x,y) &\rightarrow \mathcal{Q}_2(x)\\
\text{GenericBodyStructure}(x) \wedge \text{FeatureStateAttribute}(x,y) &\rightarrow \mathcal{Q}_3(x)\\
\text{leftRightPaired}(x) \wedge \text{hasIntrinsicAbnormalityStatus}(x,y) &\rightarrow \mathcal{Q}_4(x)\\
\text{SolidBodxyStructure}(x) \wedge \text{hasState}(x,y) &\rightarrow \mathcal{Q}_5(x)
\end{aligned}
$$

$\mathbf{G}_3$

$$
\begin{aligned}
\text{MicroOrganism}(x) \wedge \text{playsPhysiologicalRole}(x,y) &\rightarrow \mathcal{Q}_1(x)\\
\text{MicroscopicStructure}(x) \wedge \text{actsOn}(x,y) &\rightarrow \mathcal{Q}_2(x)\\
\text{Process}(x) \wedge \text{hasGoal}(x,y) &\rightarrow \mathcal{Q}_3(x)\\
\text{Feature}(x) \wedge \text{hasFeature}(x,y) &\rightarrow \mathcal{Q}_4(x)\\
\text{Process}(x) \wedge \text{hasOutcome}(x,y) &\rightarrow \mathcal{Q}_5(x)
\end{aligned}
$$

Παράρτημα Β΄

Αποδόσεις Ξένων Όρων

agent	:	πράκτορας
alignment	:	αντιστοίχιση
answer variable	:	μεταβλητή απάντησης
atomic	:	ατομικός
assertion	:	ισχυρισμός
axiom	:	αξίωμα
basic	:	βασικός
calculus	:	λογισμός
clause	:	πρόταση
combined rewriting	:	συνδυαστική επανεγγραφή
concept	:	έννοια
conclusion	:	συμπέρασμα
conjunction	:	τομή
concjunctive query	:	συζευκτικό ερώτημα
constant	:	σταθερά
constructor	:	κατασκευαστής
data integration	:	ολοκλήρωση δεδομένων
database	:	βάση δεδομένων
robistly decidable	:	εύρωστα αποφασίσιμος
deductive	:	επαγωγικός
description logic	:	περιγραφική λογική
disjunction	:	διάζευξη
distinguished variable	:	διακεκριμένη μεταβλητή
domain	:	χώρος, πεδίο
existential restriction	:	υπαρξιακός περιορισμός
existential rule	:	υπαρξιακός κανόνας
factorization	:	παραγοντοποίηση
formal logic	:	τυπική λογική
formula	:	φόρμουλα
functional term	:	συναρτησιακός όρος
fuzzy interpretation	:	ασαφής ερμηνεία
general	:	γενικός
graph	:	γράφος
ground fact	:	βασικό γεγονός

individual	:	άτομο
inconsistent	:	ασυνεπής
inclusion axiom	:	αξίωμα υπαγωγής
incremental	:	επαυξητικός
independent	:	ανεξάρτητος
inference	:	συμπερασμός
inference step	:	βήμα συμπερασμού
induction	:	επαγωγή
instance	:	στιγμιότυπο
instance relation	:	σχέση στιγμιοτύπου
interpretation	:	ερμηνεία
inverse	:	αντίστροφος
knowledge	:	γνώση
knowledge base	:	βάση γνώσης
knowledge representation	:	αναπαράσταση γνώσης
logic	:	λογική
mapping	:	αντιστοίχιση
nominal	:	ονοματική έννοια
number restriction	:	περιορισμός πληθυκότητας
object	:	αντικείμενο
ontology	:	οντολογία
ontology alignment	:	αντιστοίχιση οντολογιών
open world assumption	:	υπόθεση ανοιχτού κόσμου
program	:	πρόγραμμα
quantifier	:	ποσοδείκτης
query	:	ερώτημα
query answering	:	απάντηση ερωτημάτων
query rewriting	:	επανεγγραφή ερωτημάτων
reachable	:	προσβάσιμος
reasoner	:	μηχανή συλλογιστικής
reasoning	:	συλλογιστική
reduction	:	μείωση, αναγωγή
redundant	:	περιττός
refinement	:	τροποποίηση
reformulation	:	αναδιατύπωση
resolution	:	ανάλυση
restriction	:	περιορισμός
rewritable	:	επανεγγράψιμος
role	:	ρόλος
rule	:	κανόνας
safe	:	ασφαλής
satisfiable	:	ικανοποιήσιμος
semantics	:	σημασιολογία
substitution	:	αντικατάσταση
subsumption	:	υπαγωγή
syntax	:	συντακτικό
tableau	:	ταμπλώ

theoerm proving	:	απόδειξη θεωρημάτων
top concept	:	καθολική έννοια
top vertice	:	κόμβος κορυφής
tractable	:	βατός
transitive	:	μεταβατικός
union	:	ένωση
unqualified existential restriction	:	απροσδιόριστος υπαρξιακός περιορισμός
variable	:	μεταβλητή
view	:	όψη

Παράρτημα Γ'

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 στις ΠΛ χρησιμοποιείται στους υπαρξιακούς περιορισμούς
$\forall$	:	Ο καθολικός προσοδείκτης που στις ΠΛ χρησιμοποιείται στους περιορισμούς τιμής
$\sqsubseteq$	:	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
P^-	:	Ο αντίστροφος του ρόλου P
$\mathcal{T}$	:	Ένα σώμα ορολογίας
$\mathcal{A}$	:	Ένα σώμα ισχυρισμών
$\mathcal{I}$	:	Μια ερμηνεία (interpretation)
$\Delta^{\mathcal{I}}$	:	Ένας χώρος ερμηνείας (domain of interpretation) ο οποίος αποτελείται από ένα σύνολο αντικειμένων
$\cdot^{\mathcal{I}}$	:	Μια συνάρτηση ερμηνείας που ερμηνεύει τα στοιχεία μιας ΠΛ γλώσσας
$\models$	:	Το σύμβολο της λογικής συνεπαγωγής (entailment)
$\mathcal{G}$	:	Ένας γράφος επανεγγραφής
$\sharp\{\cdot\}$	:	Η πληθυκότητα ενός συνόλου
$\mathsf{var}(q)$	:	Το σύνολο των μεταβλητών του ερωτήματος q
$\mathsf{avar}(q)$	:	Το σύνολο των διακεκριμένων μεταβλητών του ερωτήματος q
mgu	:	Η συνάρτηση εύρεσης του μέγιστου κοινού ενοποιητή (most general unifier)
c	:	Τελεστής ασαφούς συμπληρώματος (άρνησης)
t	:	Τελεστής ασαφούς τομής (τ-νόρμα)
u	:	Τελεστής ασαφούς ένωσης (σ-νόρμα)

$\mathcal{J}$	:	Τελεστής ασαφούς συνεπαγωγής
sup	:	supremum: Το ελάχιστο άνω φράγμα ενός συνόλου αριθμών
inf	:	infimum: Το μέχιστο κάτω φράγμα ενός συνόλου αριθμών
$\bowtie$	:	Αντιπροσωπεύει μια από τις ανισσότητες $\geq, >, <$ ή $\leq$

$\square$

Παράρτημα Δ΄

Γλωσσάριο Εννοιών

Αλγόριθμος Συλλογιστικής (Reasoning algorithm) Ένας αλγόριθμος ο οποίος αποφασίζει (decides) τα προβλήματα συλλογιστικής (υπηρεσίες εξαγωγής συμπερασμάτων) μιας γλώσσας αναπαράστασης γνώσης

Αξιώμα υπαγωγής εννοιών (concept inclusion axiom) Ένα αξίωμα υπαγωγής εννοιών

Ασαφές Σύνολο (fuzzy set) Η γενίκευση ενός κλασικού (crisp) συνόλου, το οποίο περιλαμβάνει στοιχεία τα οποία ανήκουν στο ασαφές σύνολο σε κάποιο βαθμό συμμετοχής σε αντίθεση με το να ανήκουν ή όχι πλήρως

Άτομο (individual) Μια οντότητα η οποία δίνει ταυτότητα σε ένα αντικείμενο στη σύνταξη της γλώσσας και με τη χρήση του οποίου μπορούμε να δημιουργήσουμε σχέσεις στιγμιοτύπου ανάμεσα σε αυτό και μια έννοια ή ανάμεσα σε αυτό, κάποιο άλλο άτομο και έναν ρόλο.

Ατομική έννοια (ρόλος) (atomic concept (role)) Μια έννοια (ρόλος) που σε μια βάση γνώσης δεν αναλύεται σε άλλες απλούστερες, είναι δηλαδή πρωτογενής

Έννοια (concept) Ένα μοναδιαίο (unary) κατηγόρημα το οποίο αντιπροσωπεύει ένα σύνολο αντικειμένων

Ερμηνεία (interpretation) Μια μαθηματική δομή η οποία μας δίνει μια συγκεκριμένη τυπική σημασία για τα στοιχεία μιας γλώσσας και κατ' επέκταση και της οντολογίας

Ισοδυναμία εννοιών (ρόλων) (concept (role) equivalence) Ένα αξίωμα της μορφής $C_1 \equiv C_2$ ($E_1 \equiv E_2$) το οποίο δηλώνει ότι η έννοια C_1 (ρόλος E_1) και η έννοια C_2 (ρόλος E_2) είναι ισοδύναμες

Ισχυρισμός (assertion) Μία σχέση στιγμιοτύπου (ή αλλιώς αξίωμα ατόμων) ανάμεσα σε ένα άτομο και μια έννοια ή ανάμεσα σε ένα ζεύγος ατόμων και ένα ρόλο

Λογική Συνεπαγωγή (entailment) Το σύνολο (πεδίο) πάνω στο οποίο ερμηνεύουμε τη βάση γνώσης μας

Μοντέλο (model) Μία ερμηνεία η οποία ικανοποιεί ένα αξίωμα υπαγωγής, ρόλων ή ατόμου, ένα σώμα ορολογίας, ρόλων ή ισχυρισμών ή μια βάση γνώσης

Οντολογία (Ontology) Ένα σύνολο αξιωμάτων το οποίο περιγράφει τα στοιχεία (έννοιες, σχέσεις, αντικείμενα ή άτομα) και τις μεταξύ τους συσχετίσεις σε ένα συγκεκριμένο πεδίο (domain)

Ρόλος (role) Ένα δυαδικό (binary) κατηγόρημα το οποίο αντιπροσωπεύει ένα σύνολο από ζεύγη αντικειμένων

Σημασιολογία (semantics) Μια μαθηματική θεωρία με βάση την οποία περιγράφεται το πώς μπορούμε να δώσουμε ερμηνεία (σημασία) στα στοιχεία μιας γλώσσας

Σύνθετη έννοια (Έννοια Περιγραφής) (Complex concept (Concept Description)) Μία έννοια η οποία χρησιμοποιεί τις ατομικές έννοιες με σκοπό να περιγράψει περισσότερο σύνθετες έννοιες

Σώμα Ισχυρισμών (ABox) Ένα σύνολο από ισχυρισμούς

Σώμα Ορολογίας (TBox) Ένα σύνολο από αξιώματα υπαγωγής ($C_1 \sqsubseteq C_2$) και ισοδυναμίας εννοιών και ρόλων ($C_1 \equiv C_2$)

Υπαγωγή εννοιών (ρόλων) (concept (role) subsumption) Ένα αξίωμα της μορφής $C_1 \sqsubseteq C_2$ ($E_1 \sqsubseteq E_2$) το οποίο δηλώνει ότι η έννοια C_1 (ρόλος E_1) είναι υποέννοια, εξειδίκευση ή αλλιώς λιγότερο γενική (υπο-ρόλος) της έννοιας C_2 (του ρόλου E_2)

Χώρος ερμηνείας (domain of interpretation) Μια μαθηματική δομή η οποία μας δίνει μια συγκεκριμένη τυπική σημασία για τα στοιχεία μιας γλώσσας και κατ' επέκταση και της οντολογίας

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 *Proceedings of the 20th International Conference on Artificial Intelligence (AAAI)*, 2005.

[3] Yuan An, Alex Borgida, and John Mylopoulos. Discovering the semantics of relational tables through mappings. In *Journal on Data Semantics*, pages 1–32. Springer-Verlag, Berlin, Heidelberg, 2006.

[4] P.B. Andrews. *An Introduction to Mathematical Logic and Type Theory: To Truth Through Proof.* Applied Logic Series. Springer, 2002.

[5] Grigoris Antoniou and Mary-Anne Williams. *Nonmonotonic reasoning.* MIT Press, 1997.

[6] Alessandro Artale, Diego Calvanese, Roman Kontchakov, and Michael Zakharyaschev. The DL-Lite family and relations. *Journal of Artificial Intelligence Research*, 36:1–69, 2009.

[7] Th. Athanasiadis, N. Simou, G. Papadopoulos, R. Benmokhtar, K. Chandramouli, V. Tzouvaras, V. Mezaris, M. Phiniketos, Y. Avrithis, Y. Kompatsiaris, B. Huet, and E. Izquierdo. Integrating image segmentation and classification for fuzzy knowledge-based multimedia indexing. In *Proceedings of 15th International MultiMedia Modeling Conference (MMM2009)*, 2009.

[8] Franz Baader. Terminological cycles in a description logic with existential restrictions. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, pages 325–330. Morgan Kaufmann, 2003.

[9] 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.

[10] Leo Bachmair and Harald Ganzinger. *Resolution Theorem Proving*, volume 1, chapter 2, pages 19–99. Handbook of Automated Reasoning, Elsevier, Amsterdam, the Netherlands, 2001.

[11] Jean-François Baget, Michel Leclère, Marie-Laure Mugnier, and Eric Salvat. On rules with existential variables: Walking the decidability line. *Artificial Intelligence*, 175(9-10):1620–1654, 2011.

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

[13] Sean Bechhofer and Carole Goble. Description Logics and Multimedia - Applying Lessons Learnt from the GALEN Project. In *Proceedings of KRIMS 96 Workshop on Knowledge Representation for Interactive Multimedia Systems, ECAI 96*, Budapest, 1996.

[14] 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.

[15] J. I. Berman, H. H. Moore IV, and J. R. Wright. Classic and prose stories: Enabling technologies for knowledge based systems. *ATT Technical Journal*, pages 69–78, 1994.

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

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

[18] 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.

[19] Tim Berners-Lee, James Hendler, and Ora Lassila. The Semantic Web. *Scientific American*, 2001.

[20] Birte Glimm and Ian Horrocks and Boris Motik and Giorgos Stoilos. Optimising ontology classification. In *Proceedings of the 9th International Semantic Web Conference (ISWC 2010)*, volume 6496 of *LNCS*, pages 225–240, Shanghai, China, November 7–11 2010. Springer.

[21] S. Bloehdorn, K. Petridis, C. Saathoff, N. Simou, V. Tzouvaras, Y. Avrithis, S. Handschuh, Y. Kompatsiaris, S. Staab, and M. G. Strintzis. *Semantic Annotation of Images and Videos for Multimedia Analysis.* Lecture Notes in Computer Science – The Semantic Web: Research and Applications, Springer, Vol. 3532, 2005, pp. 592-607, 2005.

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

[23] Alexander Borgida and Luciano Serafini. Distributed description logics: Assimilating information from peer sources. In *Journal of Data Semantics I*, volume 1 of *Lecture Notes in Computer Science*, pages 153–184. Springer, 2003.

[24] R. Brachman and H. Levesque. *Knowledge Representation and Reasoning*. Morgan Kaufmann, 2004.

[25] 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.

[26] 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.

[27] Andrea Calì, Georg Gottlob, and Andreas Pieris. Tractable query answering over conceptual schemata. In *Proceedings of the 28th International Conference on Conceptual Modeling (ER 2009)*, pages 175–190, 2009.

[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. Ontological query answering under expressive entity-relationship schemata. *Information Systems*, 37(4):320–335, 2012.

[30] Andrea Calì, Thomas Lukasiewicz, Livia Predoiu, and Heiner Stuckenschmidt. A framework for representing ontology mappings under probabilities and inconsistency. In *Proceedings of Uncertainty Reasoning for the Semantic Web*, 2007.

[31] 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.

[32] Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, Riccardo Rosati, and Guido Vetere. Dl-lite: Practical reasoning for rich dls. In *Proc. of the 17th Int. Workshop on Description Logics (DL 2004)*, volume 104 of *CEUR Electronic Workshop Proceedings, http://ceur-ws.org/*, pages 92–99, 2004.

[33] 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.

[34] 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.

[35] S. Castano, A. Ferrara, and S. Montanelli. Matching ontologies in open networked systems: Techniques and applications. *Journal on Data Semantics (JoDS)*, V, 2005.

[36] Silvana Castano, Alfio Ferrara, Davide Lorusso, Tobias Henrik Näth, and Ralf Moeller. Mapping validation by probabilistic reasoning. In *Proceedings of the 5th European Semantic Web Conference*, LNCS. Springer Verlag, June 2008.

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

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

[39] Rada Chirkova, Alon Y. Halevy, and Dan Suciu. A formal perspective on the view selection problem. *The Very Large Data Bases Journal*, 11(3):216–237, November 2002.

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

[41] Alexandros Chortaras, Despoina Trivela, and Giorgos Stamou. Goal-oriented query rewriting for OWL 2 QL. In *24th International Workshop on Description Logics (DL 2011)*, 2011.

[42] A Collins and M Quillian. Retrieval time from semantic memory. *Journal Of Verbal Learning And Verbal Behavior*, 8(2):240–247, 1969.

[43] Calero Coral, Ruiz Francisco, and Piattini Mario. *Ontologies for Software Engineering and Software Technology.* Springer-Verlag, Berlin, Heidelberg, 2006.

[44] Jérôme David, Jérôme Euzenat, François Scharffe, and Cássia Trojahn dos Santos. The Alignment API 4.0. *Semantic Web*, 2(1):3–10, January 2011.

[45] Elena Demidova, Xuan Zhou, and Wolfgang Nejdl. A probabilistic scheme for keyword-based incremental query construction. *IEEE Transactions on Knowledge and Data Engineering*, 99, 2011.

[46] Marcos Didonet, Del Fabro, Jean Bézivin, and Patrick Valduriez. Weaving models with the eclipse amw plugin. In *Proceedings of Eclipse Modeling Symposium, Eclipse Summit Europe*, 2006.

[47] Hong-Hai Do and Erhard Rahm. Coma: a system for flexible combination of schema matching approaches. In *Proceedings of the 28th International Conference on Very Large Data Bases*, VLDB '02, pages 610–621. VLDB Endowment, 2002.

[48] Thomas Eiter, Carsten Lutz, Magdalena Ortiz, and Mantas Simkus. Query answering in description logics with transitive roles. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 759–764, 2009.

[49] Thomas Eiter, Magdalena Ortiz, Mantas Simkus, Trung-Kien Tran, and Guohui Xiao. Query Rewriting for Horn-SHIQ Plus Rules. In *Proceedings of the Association for the Advancement of Artificial Intelligence (AAAI)*, 2012.

[50] Jérôme Euzenat and Pavel Shvaiko. *Ontology matching*. Springer, 2007.

[51] Jérôme Euzenat and Petko Valtchev. Similarity-based ontology alignment in owl-lite. In *European Conference on Artificial Intelligence*, pages 333–337, 2004.

[52] FaCT++. `http://owl.man.ac.uk/factplusplus/`, 2003.

[53] M. Fitting. *First-Order Logic and Automated Theorem Proving*. Graduate Texts in Computer Science. Springer, 1996.

[54] M. R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.

[55] Matthew L. Ginsberg, editor. *Readings in nonmonotonic reasoning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1987.

[56] Fausto Giunchiglia, Aliaksandr Autayeu, and Juan Pane. S-match: An open source framework for matching lightweight ontologies. *Semantic Web*, 3(3):307–317, 2012.

[57] Fausto Giunchiglia and Pavel Shvaiko. Semantic matching. *Knowledge Engineering Review*, 18(3):265–280, 2003.

[58] Birte Glimm, Ian Horrocks, Carsten Lutz, and Uli Sattler. Conjunctive query answering for the description logic $\mathcal{SHIQ}$. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI 2007)*, 2007.

[59] Birte Glimm, Ian Horrocks, and Boris Motik. Optimized Description Logic Reasoning via Core Blocking. In Jürgen Giesl and Reiner Hähnle, editors, *Proc. of the 5th Int. Joint Conf. on Automated Reasoning (IJCAR 2010)*, volume 6173 of *LNCS*, pages 457–471, Edinburgh, UK, July 16–19 2010. Springer.

[60] Birte Glimm, Ian Horrocks, and Ulrike Sattler. Conjunctive query answering for description logics with transitive roles. In *Proceedings of the International Workshop on Description Logics (DL2006)*, 2006.

[61] Christine Golbreich, Matthew Horridge, Ian Horrocks, Boris Motik, and Rob Shearer. OBO and OWL: leveraging semantic web technologies for the life sciences. In *Proceedings of the 6th international The semantic web and 2nd Asian conference on Asian semantic web conference*, ISWC'07/ASWC'07, pages 169–182, Berlin, Heidelberg, 2007. Springer-Verlag.

[62] Christine Golbreich, Songmao Zhang, and Olivier Bodenreider. The foundational model of anatomy in OWL: Experience and perspectives. *Web Semantics*, 4(3):181–195, September 2006.

[63] John Goodwin. Experiences of Using OWL at the Ordnance Survey. In *Proceedings of OWL: Experiences and Directions*, 2005.

[64] Georg Gottlob, Giorgio Orsi, and Andreas Pieris. Ontological queries: Rewriting and optimization. In *Proceedings of the 27th International Conference on Data Engineering (ICDE 2011)*, pages 2–13, 2011.

[65] 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.

[66] Georg Gottlob, Giorgio Orsi, Andreas Pieris, and Mantas Simkus. Datalog and Its Extensions for Semantic Web Databases. In *Proceedings of the Reasoning Web*, pages 54–77, 2012.

[67] Georg Gottlob and Thomas Schwentick. Rewriting Ontological Queries into Small Nonrecursive Datalog Programs. In *Proceedings of the 24th International Workshop on Description Logics (DL 2011)*, 2011.

[68] Yuanbo Guo, Zhengxiang Pan, and Jeff Heflin. LUBM: A benchmark for OWL knowledge base systems. *Web Semantics*, 3(2-3):158–182, October 2005.

[69] Ashish Gupta, Inderpal S. Mumick, and Kenneth A. Rossl. Adapting materialized views after redefinitions. In *Proceedings of ACM SIGMOD International Conference on Management of Data*, pages 211–222, 1995.

[70] Himanshu Gupta and Inderpal Singh Mumick. Selection of views to materialize in a data warehouse. *IEEE Transactions on Knowledge Data Engineering*, 17(1):24–43, 2005.

[71] Volker Haarslev and Ralf Möller. RACER System Description. In *International Joint Conference on Automated Reasoning-01*, volume 2083, 2001.

[72] Volker Haarslev, Ralf Möller, Ragnhild Van Der Straeten, and Michael Wessel. Extended query facilities for racer and an application to software-engineering problems. In *Proceedings of the International Workshop on Description Logics (DL-2004)*, 2004.

[73] Petr Hajek. *Metamathematics of fuzzy logic.* Kluwer, 1998.

[74] Wilfrid Hodges. *A Shorter Model Theory.* Cambridge University Press, 1997.

[75] Ian Horrocks, Oliver Kutz, and Ulrike Sattler. The even more irresistible $\mathcal{SROIQ}$. In *Proceedings of the 10th International Conference on Principles of Knowledge Representation and Reasoning (KR 2006)*, pages 57–67. AAAI Press, 2006.

[76] 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.

[77] 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.

[78] 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.

[79] Bernard J. Jansen, Amanda Spink, Chris Blakely, and Sherry Koshman. Defining a session on web search engines: Research articles. *Journal of the American Society for Information Science and Technology*, 58:862–871, 2007.

[80] Bernard J. Jansen, Amanda Spink, and Jan Pedersen. A temporal comparison of altavista web searching: Research articles. *Journal of the American Society for Information Science and Technology*, 56:559–570, 2005.

[81] Jorgen Jørgensen. *A treatise of formal logic.* A Treatise of Formal Logic. Russell & Russell, 1962.

[82] Yannis Kalfoglou and W. Marco Schorlemmer. Ontology mapping: the state of the art. *Knowledge Engineering Review*, 18(1):1–31, 2003.

[83] Yiannis. Kalfoglou and Bo. Hu. Crosi mapping system (cms) results of the 2005 ontology alignment contest. In *Proceedings of the 3rd International Conference on Knowledge Capture (KCap 2005) workshop on Integrating Ontologies, Banff, Canada*, Oct 2005.

[84] Yevgeny Kazakov. *Saturation-Based Decision Procedures for Extensions of the Guarded Fragment.* PhD thesis, Universität des Saarlandes, Saarbrücken, Germany, March 2006.

[85] 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.

[86] G. J. Klir and B. Yuan. *Fuzzy Sets and Fuzzy Logic: Theory and Applications.* Prentice-Hall, 1995.

[87] 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.

[88] Adila Krisnadhi and Carsten Lutz. Data complexity in the EL family of description logics. In *Proceedings of the International Workshop on Description Logics*, 2007.

[89] 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.

[90] Markus Krötzsch and Sebastian Rudolph. Conjunctive queries for EL with composition of roles. In Diego Calvanese, Enrico Franconi, Volker Haarslev, Domenico Lembo, Boris Motik, Anni-Yasmin Turhan, and Sergio Tessaris, editors, *Proceedings of the 2007 International Workshop on Description Logics (DL2007), Brixen-Bressanone, near Bozen-Bolzano, Italy, 8-10 June, 2007*, volume 250 of *CEUR Workshop Proceedings.* CEUR-WS.org, 2007.

[91] 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.

[92] Clarence Irving Lewis and Cooper Harold Langford. *Symbolic logic.* Century philosophy series. The Century co., 1932.

[93] 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.

[94] 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.

[95] X. Ma. *Ontology Spectrum for Geological Data Interoperability.* ITC publication. 2011.

[96] David M. Mark, Barry Smith, and Barbara Tversky. Ontology and geographic objects: an empirical study of cognitive categorization. In *Lecture Notes in Computer Science*, pages 283–298, 1999.

[97] Deborah L. McGuinness. Conceptual modeling for distributed ontology environments. In *Proceedings of the Linguistic on Conceptual Structures: Logical Linguistic, and Computational Issues*, ICCS '00, pages 100–112, London, UK, UK, 2000. Springer-Verlag.

[98] Deborah L. McGuinness and Jon R. Wright. Conceptual modelling for configuration: A description logic-based approach. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 12(4):333–344, September 1998.

[99] Carlo Meghini, Fabrizio Sebastiani, and Umberto Straccia. A model of multimedia information retrieval. *Journal of the ACM*, 48(5):909–970, 2001.

[100] Elliot Mendelson. *Introduction to Mathematical Logic.* Chapman & Hall/CRC, 1997.

[101] Mukesh Mohania. Avoiding re-computation: View adaptation in data warehouses. In *Proceedings of 8th International Database Workshop*, pages 151–165, 1997.

[102] Ralf Möller, Bernd Neumann, and Michael Wessel. Towards computer vision with description logics-some recent progress. In *Workshop on Integration of Speech and Image Understanding*, pages 101–115, 1999.

[103] Boris Motik. *Reasoning in Description Logics using Resolution and Deductive Databases.* PhD thesis, Universität Karlsruhe (TH), Karlsruhe, Germany, January 2006.

[104] Boris Motik, Bernardo Cuenca Grau, Ian Horrocks, Zhe Wu, Achille Fokoue, and Carsten Lutz (Editors). OWL 2 Web Ontology Language Profiles, 2009.

[105] 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.

[106] Boris Motik, Rob Shearer, and Ian Horrocks. Hypertableau Reasoning for Description Logics. *Journal of Artificial Intelligence Research*, 36:165–228, 2009.

[107] Bernd Neumann and Ralf Möller. On Scene Interpretation with Description Logics. In H.I. Christensen and H.-H. Nagel, editors, *Cognitive Vision Systems: Samping the Spectrum of Approaches*, number 3948 in LNCS, pages 247–278. Springer, 2006.

[108] Giorgio Orsi and Andreas Pieris. Optimizing Query Answering under Ontological Constraints. *The Proceedings of the Very Large Data Bases Endowment*, 4(11):1004–1015, 2011.

[109] Magdalena Ortiz, Diego Calvanese, and Thomas Eiter. Characterizing data complexity for conjunctive query answering in expressive description logics. In *In Proc. of AAAI 2006*, 2006.

[110] 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.

[111] Jeff Z. Pan, Giorgos Stamou, Giorgos Stoilos, and Edward Thomas. Expressive querying over fuzzy DL-Lite ontologies. In *Proceedings of the International Workshop on Description Logics (DL 2007)*, 2007.

[112] Christos H. Papadimitriou. *Computational complexity.* Academic Internet Publ., 2007.

[113] Georgios Papadopoulos, Thanos Athanasiadis, Nikolaos Simou, Rachid Benmokhtar, Krishna Chrandramouli, Vassilis Tzouvaras, Vassilis Mezaris, Marios Phiniketos, Yannis Avrithis, and Ioannis Kompatsiaris. Combining Segmentation and Classification Techniques for Fuzzy Knowledge-based Semantic Image Annotation. In *Proceedings of 3rd international conference on Semantics And digital Media Technologies (SAMT), Koblenz, December 2008*, 2008.

[114] Greg Pass, Abdur Chowdhury, and Cayley Torgeson. A picture of search. In *Proceedings of the 1st international conference on Scalable information systems (InfoScale 06)*. ACM, 2006.

[115] 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/`.

[116] 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.

[117] 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.

[118] Kosmas Petridis, Stephan Bloehdorn, Carsten Saathoff, Nikolaos Simou, Stamatia Dasiopoulou, Vassilis Tzouvaras, Siegfried Handschuh, Yannis Avrithis, Ioannis Kompatsiaris, and Steffen Staab. Knowledge representation and semantic annotation of multimedia content. *IEE Proceedings on Vision Image and Signal Processing, Special issue on Knowledge-Based Digital Media Processing*, 153(3):255–262, 2006.

[119] Amir Pnueli. The temporal logic of programs. *IEEE Annual Symposium on Foundations of Computer Science*, 0:46–57, 1977.

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

[121] Erhard Rahm and Philip A. Bernstein. A survey of approaches to automatic schema matching. *The Very Large Data Bases Journal*, 10(4):334–350, December 2001.

[122] Rob Raskin and Michael Pan. Semantic web for earth and environmental terminology (sweet. In *Proceedings of the Workshop on Semantic Web Technologies for Searching and Retrieving Scientific Data*, 2003.

[123] Wolfgang Rautenberg. *A Concise Introduction to Mathematical Logic.* Springer Science+Business Media, Incorporated, 2006.

[124] Alan L. Rector and Ian Horrocks. Experience building a large, re-usable medical ontology using a description logic with transitivity and concept inclusions. In *Proceedings Workshop on Ontological Engineering, AAAI Spring Symposium, Stanford CA.*, pages 100–107. Hanley and Belfus, Inc., Philadelphia, PA, 1997.

[125] John A. Robinson. A machine-oriented logic based on the resolution principle. *Journal ACM*, 12(1):23–41, January 1965.

[126] John A. Robinson and Andrei Voronkov. *Handbook of automated reasoning. 1.* Handbook of Automated Reasoning. Elsevier Science Limited, 2001.

[127] Mariano Rodriguez-Muro and Diego Calvanese. High performance query answering over DL-Lite ontologies. In *Proceedings of the 13th International Conference Principles of Knowledge Representation and Reasoning, KR*, 2012.

[128] Mariano Rodriguez-Muro, Lina Lubyte, and Diego Calvanese. Realizing ontology based data access: A plug-in for protégé. In *Proceedings of the 24th International Conference on Data Engineering Workshops, ICDE 2008, April 7-12, 2008, Cancún, México*, pages 286–289, 2008.

[129] Riccardo Rosati. On conjunctive query answering in el. In *Proceedings of the 2007 International Workshop on Description Logics (DL2007)*, 2007.

[130] Riccardo Rosati. Prexto: Query rewriting under extensional constraints in DL-Lite. In *Proceedings of the 9th Extended Semantic Web Conference*, pages 360–374, 2012.

[131] Riccardo Rosati. Query rewriting under extensional constraints in DL-Lite. In *Proceedings of the International Workshop on Description Logics, DL-2012*, 2012.

[132] Riccardo Rosati and Alessandro Almatelli. Improving query answering over DL-Lite ontologies. In *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR-10)*, 2010.

[133] Ulrike Sattler. Description logics for the representation of aggregated objects. In *Proceedings of the 14th European Conference on Artificial Intelligence. IOS*, pages 239–243. Press, 2000.

[134] M.S. Schauß and G. Smolka. Attributive concept descriptions with complements. *Artificial Intelligence*, 48(1):1–26, 1991.

[135] H. Scholz. *Geschichte der Logik*. Geschichte der Philosophie in Längsschnitten. Juncker u. Dünnhaupt, 1931.

[136] Stefan Schulz, Ronald Cornet, and Kent Spackman. Consolidating SNOMED CT's ontological commitment. *Applied Ontology*, 6(1):1–11, January 2011.

[137] Nigel Shadbolt, Tim Berners-Lee, and Wendy Hall. The Semantic Web Revisited. *IEEE Intelligent Systems*, 21(3):96–101, 2006.

[138] J.R. Shoenfield. *Mathematical Logic*. Addison-Wesley, 1967.

[139] Pavel Shvaiko and Jérôme Euzenat. Ontology Matching: State of the Art and Future Challenges. *IEEE Transactions on Knowledge Data Engineering*, 25(1):158–176, 2013.

[140] Amandeep S. Sidhu, Tharam S. Dillon, Elizabeth Chang, and Baldev S. Sidhu. Protein ontology development using owl. In Bernardo Cuenca Grau, Ian Horrocks, Bijan Parsia, and Peter F. Patel-Schneider, editors, *OWLED*, volume 188 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2005.

[141] N. Simou, Th. Athanasiadis, and S. Kollias. An architecture for multimedia analysis and retrieval based on fuzzy description logics. 2nd K-Space PhD Students Workshop, Paris, France, July 2008, 2008.

[142] N. Simou, Th. Athanasiadis, G. Stoilos, and S. Kollias. Image indexing and retrieval using expressive fuzzy description logics. *Signal, Image and Video Processing*, 2(4):321–335, 2008.

[143] Nikolaos Simou, Thanos Athanasiadis, Vassilis Tzouvaras, and Stefanos Kollias. Multimedia reasoning with f-shin. 2nd International Workshop on Semantic Media Adaptation and Personalization, London, December 17-18, 2007, London, United Kingdom (SMAP 2007), 2007.

[144] Nikolaos Simou and Stefanos Kollias. Fire : A fuzzy reasoning engine for impecise knowledge. In *K-Space PhD Students Workshop, Berlin, Germany*, 2007.

[145] Nikos Simou, Giorgos Stoilos, and Giorgos Stamou. Storing and querying fuzzy knowledge in the semantic web using fire. In Fernando Bobillo, Paulo Cesar G. da Costa, Claudia d'Amato, and Nicola Fanizzi et al., editors, *Uncertainty Reasoning for the Semantic Web II, ISWC International Workshops URSW 2008-2010 Held at ISWC and UniDL 2010 Held at FLoC, Revised Selected Papers*, pages 158–176, 2013.

[146] 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.

[147] Thoralf Skolem. *Logisch-kombinatorische Untersuchungen über die Erfüllbarkeit oder Beweisbarkeit mathematischer Sätze nebst einem Theoreme ber dichte Mengen.* Skrifter utgitt av Videnskapsselskapet i Kristiania. 1920.

[148] Raymond Smullyan. *First-Order Logic.* Dover books on advanced mathematics. DOVER PUBN Incorporated, 1995.

[149] Giorgos Stamou, Despoina Trivela, and Alexandros Chortaras. Progressive semantic query answering. In *Scalable Semantiv Web Knowledge Base Systems Workshop (SSWS 2010), Shanghai, China*, 2010.

[150] Robert Stevens, Mikel Egaña Aranguren, Katy Wolstencroft, Ulrike Sattler, Nick Drummond, Matthew Horridge, and Alan L. Rector. Using owl to model biological knowledge. *International Journal of Man-Machine Studies*, 65(7):583–594, 2007.

[151] Giorgos Stoilos, Nikolaos Simou, Giorgos Stamou, and Stefanos Kollias. Uncertainty and the semantic web. *IEEE Intelligent Systems*, 21(5):84–87, 2006.

[152] Giorgos Stoilos, Giorgos Stamou, and Jeff Z. Pan. Handling imprecise knowledge with fuzzy description logics. In *Proceedings of the International Workshop on Description Logics (DL 2006), Lake District, UK*, 2006.

[153] Giorgos Stoilos, Giorgos Stamou, Vassilis Tzouvaras, Jeff Z. Pan, and Ian Horrocks. A fuzzy description logic for multimedia knowledge representation. In *Proc. of the International Workshop on Multimedia and the Semantic Web*, 2005.

[154] Giorgos Stoilos, Giorgos Stamou, Vassilis Tzouvaras, Jeff Z. Pan, and Ian Horrocks. Reasoning with very expressive fuzzy description logics. *Journal of Artificial Intelligence Research*, 30(5):273–320, 2007.

[155] Giorgos Stoilos, Giorgos B. Stamou, Vassilis Tzouvaras, Jeff Z. Pan, and Ian Horrocks. The fuzzy description logic f-shin. In *International Semantic Web Conference, Uncertainty Reasoning for the Semantic Web*, pages 67–76, 2005.

[156] Umberto Straccia. A fuzzy description logic. In *Association for the Advancement of Artificial Intelligence (AAAI)*, pages 594–599, 1998.

[157] Umberto Straccia. Reasoning within fuzzy description logics. *Journal of Artificial Intelligence Research*, 14:137–166, 2001.

[158] Umberto Straccia. Answering vague queries in fuzzy DL-Lite. In *Proceedings of the 11th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, (IPMU-06)*, pages 2238–2245, 2006.

[159] Jie Tang, Juanzi Li, Bangyong Liang, Xiaotong Huang, Yi Li, and Kehong Wang. Using bayesian decision for ontology mapping. *Web Semant.*, 4(4):243–262, December 2006.

[160] Thanh Tran, Philipp Cimiano, Sebastian Rudolph, and Rudi Studer. Ontology-based interpretation of keywords for semantic search. In *Proceedings of the International Semantic Web Conference (ISWC 2007)*, volume 4825, pages 523–536, 2007.

[161] Eleni Tsalapati, Giorgos Stoilos, Giorgos Stamou, and George Koletsos. Query rewriting under ontology contraction. In *Proceedings of the 6th International Conference on Web Reasoning and Rule Systems (RR 2012)*, pages 172–187, 2012.

[162] Dimitry Tsarkov and Ian Horrocks. Fact++ description logic reasoner: system description. In *Proceedings of the Third international joint conference on Automated Reasoning*, IJCAR'06, pages 292–297, Berlin, Heidelberg, 2006. Springer-Verlag.

[163] Paraskevi Tzouveli, Nikolaos Simou, Giorgos Stamou, and Stefanos Kollias. Semantic classification of byzantine icons. *IEEE Intelligent Systems*, 24(2):35–43, 2009.

[164] Alexandros Valarakos, Vassilis Spiliopoulos, and George A. Vouros. *AUTOMS-F: A Framework for the Synthesis of Ontology Mapping Methods*, pages 45–60. Networked Knowledge – Networked Media Integrating Knowledge Management, New Media Technologies and Semantic Systems, 2009.

[165] Moshe Y. Vardi. The complexity of relational query languages (extended abstract). In *Proceedings of the fourteenth annual ACM symposium on Theory of computing*, STOC '82, pages 137–146, New York, NY, USA, 1982. ACM.

[166] 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.

[167] Michal Walicki. Mathematical logic - an introduction. `http://www.ii.uib.no/~michal/und/i227/book/book.pdf`, 2005. Department of Informatics, University of Bergen.

[168] Lotfi A. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.

[169] Gideon Zenz, Xuan Zhou, Enrico Minack, Wolf Siberski, and Wolfgang Nejdl. From keywords to semantic queries-incremental query construction on the semantic web. *Journal of Web Semantics*, 7:166–176, 2009.

Κατάλογος Δημοσιεύσεων

T. Venetis, G. Stoilos, G. Stamou and S. Kollias, f-DLPs: Extending Description Logic Programs with Fuzzy Sets and Fuzzy Logic, In *Proceedings of the IEEE International Conference on Fuzzy Systems (Fuzz-IEEE' 07)*, London, 2007.

A. Ferrara, D. Lorusso, G. Stamou, G. Stoilos, V. Tzouvaras, T. Venetis, Resolution of conflicts among ontology mappings: a fuzzy approach, In *Proceedings of the International Workshop on Ontology Matching (OM2008)*, 2008.

T. Venetis, G. Stoilos, G. Stamou, Query Rewriting Under Query Extensions for OWL 2 QL Ontologies, In *Proceedings of the 7th International Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS2011)*, 2011.

T. Venetis, G. Stoilos, G. Stamou, Incremental Query Rewriting for OWL 2 QL, In *Proceedings of the 25th International Workshop on Description Logics (DL2012)*, 2012.

T. Venetis, G. Stoilos, G. Stamou, Query Extensions and Incremental Query Rewriting for OWL 2 QL, In *Journal on Data Semantics*, accepted for publication 2013.

T. Venetis, G. Stoilos, G. Stamou, Query Rewriting Under Query Refinements, In *Knowledge-Based Systems*, accepted for publication 2013.

□

Ευρετήριο Όρων