Εθνικο Μετσοβιο Πολυτεχνειο

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧ. ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Συστήματα και Τεχνολογίες Γνώσης

2η ΑΣΚΗΣΗ

Αχλάτης Στέφανος-Σταμάτης (03116149)

<el16149@central.ntua.gr>

Ιούνιος 2020

Θεμα 1ο

Αρχικά για να χρησιμοποιήσουμε τον αλγοριθμο δομική υπαγωγης για την FLo πρέπει να μετατρεψουμε τις C_1, C_2 σε κανονική μορφη

▸ Μετατροπη της C_1 σε κανονικη μορφη, οπου:

$C_1 \equiv \forall r. A \sqcap D \sqcap \forall r. \forall r. E \sqcap \forall r. B \sqcap E \sqcap \forall r. (A \sqcap B) \sqcap \forall r. \forall r. \forall s. D$

Βήμα 1 (προσεταιριστικη ιδιοτητα του τελεστη $\sqcap$) δεν τροποποιεί την έννοια

Βήμα 2 (διαχωρισμος εννοιών και ρολων), η C_1 γινεται:

$C_1 \equiv D \sqcap E \sqcap \forall r. A \sqcap \forall r. B \sqcap \forall r. (A \sqcap B) \sqcap \forall r. \forall r. E \sqcap \forall r. \forall r. \forall s. D$

Βήμα 3 (ταυτοδυναμία του τελεστή $\sqcap$) δεν τροποποιεί την έννοια

Βημα 4 (επιμεριστικοτητα του τελεστη $\forall$), η C_1 γίνεται:

$C_1 \equiv D \sqcap E \sqcap \forall r. (A \sqcap B \sqcap (A \sqcap B) \sqcap \forall r. E \sqcap \forall r. \forall s. D)$

Στην συνεχεια η διαδικασια επαναλαμβανεται για την έννοια

$C_1' \equiv A \sqcap B \sqcap (A \sqcap B) \sqcap \forall r. E \sqcap \forall r. \forall s. D$

Βήμα 1 (προσεταιριστικοτητα του τελεστη $\sqcap$), η εννοια C_1' γινεται:

$C_1' \equiv A \sqcap B \sqcap A \sqcap B \sqcap \forall r. E \sqcap \forall r. \forall s. D$

Βήμα 2 (διαχωρισμος εννοιων και ρολων) δεν τροποποιει την έννοια

Βημα 3 (ταυτοδυναμια του τελεστη $\sqcap$), η C_1' γινεται:

$C_1' \equiv A \sqcap B \sqcap \forall r. E \sqcap \forall r. \forall s. D$

Βήμα 4 (επιμεριστικοτητα του τελεστη $\forall$), η C_1 γίνεται:

$C_1' \equiv A \sqcap B \sqcap \forall r. (E \sqcap \forall s. D)$

Εφόσον αυτη είναι σε κανονική μορφη η διαδικασια τερματιζει και τελικα εχουμε:

$C_1 \equiv D \sqcap E \sqcap \forall r. (A \sqcap B \sqcap \forall r. (E \sqcap \forall s. D))$

Μετατροπη της C_2 σε κανονική μορφη όπου:
$C_2 \equiv \forall r. \forall r. (E \sqcap B) \sqcap \forall r. \forall r. \forall s. (D \sqcap A) \sqcap E$

Βήμα 1 (προσεταιριστικη ιδιοτητα του τελεστη $\sqcap$) δεν τροποποιειτην εννοια
Βήμα 2 (διαχωρισμος εννοιώνκαι ρόλων), η C_2 γίνεται:
$C_2 \equiv E \sqcap \forall r. \forall r. (E \sqcap B) \sqcap \forall r. \forall r. \forall s. (D \sqcap A)$
Βήμα 3 (ταυτοδυναμια του τελεστη $\sqcap$) δεν τροποποιει την εννοια
Βήμα 4 (επιμεριστικότητα του τελεστη $\forall$), η C_2 γινεται:
$C_2 \equiv E \sqcap \forall r. (\forall r. (E \sqcap B) \sqcap \forall r. \forall s (D \sqcap A))$

Στην συνεχεια η διαδικασια επαναλαμβανεται για την έννοια
$C_2' \equiv \forall r. (E \sqcap B) \sqcap \forall r. \forall s (D \sqcap A)$

Βημα 1 (προσεταιριστικοτητα του τελεστη $\sqcap$) δεν τροποποιεί την έννοια
Βήμα 2 (διαχωρισμος εννοιων και ρολων) δεν τροποποιεί την έννοια
Βήμα 3 (ταυτοδυναμια του τελεστη $\sqcap$) δεν τροποποιεί την έννοια
Βημα 4 (επιμεριστικοτητα του τελεστη $\forall$), η C_2' γίνεται:
$C_2' \equiv \forall r. (E \sqcap B \sqcap \forall s. (D \sqcap A)$

Εφοσου αυτή ειναι σε κανονικη μορφη η διαδικασια τερματιζει και τελικά εχουμε:

$C_2 \equiv E \sqcap \forall r. (\forall r. (E \sqcap B \sqcap \forall s (D \sqcap A))$
που ειναι ισοδυναμο με το
$C_2 \equiv E \sqcap \forall r. \forall r. (E \sqcap B \sqcap \forall s. (D \sqcap A))$

Και τωρα εχοντας φερει τα C_1, C_2 σε κανονικη μορφη μπορουμε να χρησιμοποιήσουμε τον αλγοριθμο δομικης υπαγωγης για την FLo.

- $C_1 \equiv D \sqcap E \sqcap \forall r.(A \sqcap B \sqcap \forall r.(E \sqcap \forall s.D))$
- $C_2 \equiv E \sqcap \forall r.\forall r.(E \sqcap B \sqcap \forall s.(D \sqcap A))$

- <u>$1^{η}$ κλήση του Αλγορίθμου</u>

Εκτελούμε τη συνάρτηση atom() για την έννοια C_1 η οποία δημιουργεί το σύνολο $NC_1 = \{D, E\}$, ενώ για την έννοια C_2 δημιουργεί το σύνολο $NC_2 = \{E\}$.
Εκτελούμε την συνάρτηση forall-roles() για την έννοια C_1 η οποία δημιουργεί το σύνολο $RC_1 = \{r\}$ ενώ για την έννοια C_2 δημιουργεί το σύνολο $RC_2 = \{r\}$
Εκτελούμε το for loop για έλεγχο της συνθήκης $NC_2 \subseteq NC_1$
Επειδή η συνθήκη ικανοποιείται αφού το E από NC_2 αντιστοιχίζεται στο E από το NC_1 συνεχίζουμε στο $2^{ο}$ for loop όπου για το r από το RC_2 ορίζουμε τα $X := role\text{-}restr(C_1, r) \equiv A \sqcap B \sqcap \forall r.(E \sqcap \forall s.D)$
και $Y := role\text{-}restr(C_2, r) \equiv \forall r.(E \sqcap B \sqcap \forall s.(D \sqcap A))$
και κάνουμε αναδρομική κλήση του αλγορίθμου με input τα X, Y.

- <u>$2^{η}$ κλήση του Αλγορίθμου</u>

Εκτελούμε τη συνάρτηση atom() για την έννοια C_1 (το X) η οποία δημιουργεί το σύνολο $NC_1 = \{A, B\}$, ενώ για την έννοια C_2 (το Y) δημιουργεί το σύνολο $NC_2 = \{\}$
Εκτελούμε την συνάρτηση forall-roles() για την έννοια C_1 η οποία δημιουργεί το σύνολο $RC_1 = \{r\}$ ενώ για την έννοια C_2 δημιουργεί το σύνολο $RC_2 = \{r\}$
Εκτελούμε το πρώτο for loop για έλεγχο της συνθήκης $NC_2 \subseteq NC_1$
Επειδή η συνθήκη ικανοποιείται αφού το NC_2 είναι το κενό σύνολο συνεχίζουμε στο $2^{ο}$ for loop όπου για το r από το RC_2 ορίζουμε τα $X := role\text{-}restr(C_1, r) \equiv E \sqcap \forall s.D$
και $Y := role\text{-}restr(C_2, r) \equiv E \sqcap B \sqcap \forall s D \sqcap A$.
και κάνουμε αναδρομική κλήση του αλγορίθμου με input τα X, Y.

3η κληση του Αλγοριθμου

Εκτελουμε τη συναρτηση atom() για την εννοια C_1 (το X) η οποια δημιουργει το συνολο $NC_1 = \{E\}$, ενω για την εννοια C_2 (το Y) δημιουργει το συνολο $NC_2 = \{E, B\}$.
Εκτελουμε την συναρτηση forall-roles() για την εννοια C_1 η οποια δημιουργει το συνολο $RC_1 = \{S\}$ ενω για την εννοια C_2 δημιουργει το συνολο $RC_2 = \{S\}$.
Εκτελουμε το πρωτο for loop για ελεγχο της συνθηκης $NC_2 \subseteq NC$
Η συνθηκη δεν ικανοποιειται, και ετσι επιστρεφεται NO.

Ετσι, αναδρομικα ο αλγοριθμος μας τερματιζει επιστρεφοντας NO

- Ετσι, ειδαμε οτι ο αλγοριθμος δομικης υπαγωγης του FL_0 υπεδειξε οτι η υπαγωγη $C_1 \sqsubseteq C_2$ δεν ικανοποιειται.

Θεμα 2ο

α) Να ελεγχθει με βαση τον αλγοριθμο tableau για την ALC αν ισχυει το $\neg A \sqsubseteq \exists R.B$ μβς $\mathcal{T} = \{\forall R.\neg B \sqsubseteq A\}$

Θελουμε να δουμε αν ισχυει το $\neg A \sqsubseteq \exists R.B$, το οποίο είναι ενα προβλημα υπαγωγής εννοιων. Το προβλημα μπορει να μετασχηματισει στο προβλημα ελεγχου ικανοποιησιμοτητας της εξής έννοιας:

$C \equiv \neg A \sqcap \neg(\exists R.B)$.

Για να εφαρμοσουμε τον αλγοριθμο tableau πρεπει η C να είναι σε KMA, επομενως πρεπει να φερουμε την C σε KMA. με τον γνωστο Αλγοριθμο:

Βημα 1 (Εφαρμογη του κανονα $\neg(\exists r.A) \equiv \forall r.\neg A$) και η C γινεται:

$C \equiv \neg A \sqcap \forall R.\neg B$

(Βημα 2) Επιτυχης ελεγχος οτι η C ειναι σε KMA, αρα τερματιζουμε

Αρα η $C \equiv \neg A \sqcap \forall R.\neg B$

Πριν εφαρμοσουμε τον αλγοριθμο tableau πρεπει πρωτα να κάνουμε απαλοιφη του TBox (TBox elimination). Αυτο θα το επιτυχουμε με την μεθοδος της εσωτερικευσης. Ετσι πρεπει να προστεθει για καθε ατομο του κοσμου γνωσης ο ισχυρισμος $\neg(\forall R.\neg B) \sqcup A$ ο οποιος ισοδυναμει με τον ισχυρισμο $\exists R.B \sqcup A$.

Ο τελευταιος ισχυρισμος ειναι σε KMA.

Ξεκινοντας τον αλγοριθμο tableau, η δομη tableau θα αρχικοποιηθει στην ακόλουθη τιμη:

$S_0 = \{\{C(\alpha), (\exists R.B \sqcup A)(\alpha)\}\}$ ή ισοδυναμα

$S_0 = \{\{(\neg A \sqcap \forall R.\neg B)(\alpha), (\exists R.B \sqcup A)(\alpha)\}\}$

Στην συνεχεια ο ελεγχος θα περασει στον βροχο repeat που μετασχηματιζει το S_0, οποτε με εκτελεση της συναρτησης applyRules() στο S_0, θα εφαρμοσει ο κανονας $\sqcap$ στο $C(\alpha)$, μια φορα και το δομη ταμπλο θα γινει:

$S_1 = \{ \{ C(\alpha), \neg A(\alpha), (\forall R^{\neg} B)(\alpha), (\exists R.B \sqcup A)(\alpha) \} \}$

Στο επόμενο βήμα του βρόγχου repeat, θα εκτελέσει η applyRules() στο S_1 και εφαρμόζεται ο κανόνας $\sqcup$ στον ισχυρισμό $(\exists R.B \sqcup A)(\alpha)$ οπότε η δομή παραπάνω γίνεται:

$$S_2 = \{ \{ C(\alpha), \neg A(\alpha), (\forall R^{\neg} B)(\alpha), (\exists R B \sqcup A)(\alpha), (\exists R.B)(\alpha) \}, \{ C(\alpha), \neg A(\alpha), (\forall R^{\neg} B)(\alpha), (\exists R B \sqcup A)(\alpha), A(\alpha) \} \}$$

Στο επόμενο βήμα του βρόγχου repeat, θα εκτελέσει η applyRules() στο S_2 και εφαρμόζεται ο κανόνας $\exists$ στον ισχυρισμό $(\exists R.B)(\alpha)$ οπότε το S_2 γίνεται (στο πρώτο A-Box):

$$S_3 = \{ \{ C(\alpha), \neg A(\alpha), (\forall R^{\neg} B)(\alpha), (\exists R B \sqcup A)(\alpha), (\exists R.B)(\alpha), R(\alpha, b), B(b), \exists R.(B \sqcup A)(b) \}, \{ C(\alpha), \neg A(\alpha), (\forall R^{\neg} B)(\alpha), (\exists R B \sqcup A)(\alpha), A(\alpha) \} \}$$

Στο επόμενο βήμα του βρόγχου repeat θα εκτελέσει η applyRules() στο S_3 και εφαρμόζεται ο κανόνας-$\forall$ στο οποίο ισχυρισμό $(\forall R^{\neg} B)(\alpha)$ και το δομή παραπάνω γίνεται:

$$S_4 = \{ \{ C(\alpha), \neg A(\alpha), (\forall R^{\neg} B)(\alpha), (\exists R B \sqcup A)(\alpha), (\exists R.B)(\alpha), R(\alpha, b), B(b), \exists R(B \sqcup A)(b), \neg B(b) \}, \{ C(\alpha), \neg A(\alpha), (\forall R^{\neg} B)(\alpha), (\exists R B \sqcup A)(\alpha), A(\alpha) \} \}$$

Στο επόμενο βήμα του βρόγχου repeat θα εκτελέσει η applyRules() στο S_4 και εφαρμόζεται ο κανόνας-$\sqcup$ στον ισχυρισμό $(\exists R B \sqcup A)(b)$ οπότε το S_4 γίνεται:

$S5 = \{ \{ C(\alpha), \neg A(\alpha), (\forall R.\neg B)(\alpha), (\exists R.B \sqcup A)(\alpha), \exists R.B(\alpha),$
$R(\alpha,b), B(b), (\exists R.B \sqcup A)(b), \neg B(b), (\exists R.B)(b) \},$
$\{ C(\alpha), \neg A(\alpha), \forall R.\neg B(\alpha), (\exists R.B \sqcup A)(\alpha), \exists R.B(\alpha),$
$R(\alpha,b), B(b), (\exists R.B \sqcup A)(b), \neg B(b), A(b) \},$
$\{ C(\alpha), \neg A(\alpha), (\forall R \neg B)(\alpha), (\exists R.B \sqcup A)(\alpha), A(\alpha) \} \}$

Λογω της μεθοδου της εσωτερικευσης μολις προσθεσαμε το νεο ατομο b, προσθεσαμε και τον ισχυρισμο $(\exists R.B \sqcup A)(b)$ ο οποιος ομως "εσπασε" και μας εδωσε τον ισχυρισμο $(\exists R.B)(b)$, ο οποιος ομως με τη σειρα του μπορει να δωσει νεο ατομο με τον κανονα-$\exists$ και να επαναληφθει η διαδικασια που εγινε απο το Sa μεχρι το S4 και να μην τερματισει ο αλγοριθμος ποτε. Χαρη ομως στην τεχνικη του subset blocking ο αλγοριθμος τερματιζει το συγκεκριμενο κομματι που εδω μιας και ο μονος κανονας που μπορει να εφαρμοσει πλεον ειναι αυτος που αναφερθηκε (κανονας-$\exists$) ο οποιος ομως εισαγει ενα νεο ατομο του οποιου οι ιδιοτητες ειναι υπερσυνολο ενος προγονου του.

Μετα το repeat του αλγοριθμου μας, εχουμε 3 ABoxes

$A_1 = \{ C(\alpha), \neg A(\alpha), (\forall R.\neg B)(\alpha), (\exists R.B \sqcup A)(\alpha), \exists R.B(\alpha),$
$R(\alpha,b), B(b), (\exists R.B \sqcup A)(b), \neg B(b), (\exists R.B)(b) \}$
$A_2 = \{ C(\alpha), \neg A(\alpha), \forall R.\neg B(\alpha), (\exists R.B \sqcup A)(\alpha), \exists R.B(\alpha),$
$R(\alpha,b), B(b), (\exists R.B \sqcup A)(b), \neg B(b), A(b) \}$
$A_3 = \{ C(\alpha), \neg A(\alpha), (\forall R \neg B)(\alpha), (\exists R.B \sqcup A)(\alpha), A(\alpha) \}$

Τωρα ελεγχουμε αν τα ABoxes ειναι ελευθερα αντιφασεων με αρχικο check το A1 μεσω της clashFree(A1) το A1 ομως δεν ειναι ελευθερο αντιφασεων λογω των ισχυρισμων B(b) και ¬B(b). Αρα η clashFree(A1) επιστρεφει FALSE.

Επομενως θα εκτελεσει ξανα ο βρογχος for, αυτη τη φορα για το A2, και θα εχουμε κληση της clashFree(A_2). Και αυτη ομως θα επιστρεψει FALSE λόγω της αντιφασης που εντοπιζεται στο A2 μεταξυ των ισχυρισμων B(b) και ¬B(b) ξανα.

Τέλος, θα εχουμε μια ακομη επανάληψη του βρογχου for, αυτη τη φορα για το A3, ομως και αυτη θα επιστρεψει False. επειδη εντοπιζεται αντιφαση στο A3 λογω των A(α) και ¬A(α).

Έτσι, ο βρογχος for της γραμμης 7 θα τερματισει.
Επομενως, ο ελεγχος θα περασει στην γραμμη 12, οπου η tableauSAT(C) θα επιστρεψει NO.
Άρα, η εννοια C δεν ειναι ικανοποιησιμη, και αρα <u>η αρχικη υπαγωγη ισχυει για καθε μοντελο του TBox</u>.

β) Εχουμε να εφαρμοσουμε τον αλγοριθμο tableau με δεδομενο ABox και TBox.
Ετσι πρεπει να εφαρμοσουμε TBox elimination, και πρεπει να προσθεσουμε για καθε ατομο του κοσμου τον ισχυρισμο: $(\neg A \sqcup C) \sqcap (\neg C \sqcup \exists R.D) \sqcap (\neg D \sqcup \neg A) \sqcap (\neg C \sqcup \forall R.A)$ που ειναι σε KMA και η δομη tableau θα εχει αρχικα το δεδομενο ABox και εφαρμογη του προηγουμενα κανονα για τα ατομα α και β.

$S_0 = \{ A_1 \}$

οπου $A_1 = \{ A(\alpha), R(\alpha, b), ((\neg A \sqcup C) \sqcap (\neg C \sqcup \exists R.D) \sqcap (\neg D \sqcup \neg A) \sqcap (\neg C \sqcup \forall R.A))(\alpha), ((\neg A \sqcup C) \sqcap (\neg C \sqcup \exists R.D) \sqcap (\neg D \sqcup \neg A) \sqcap (\neg C \sqcup \forall R.A))(b) \}$

- Εφαρμογη Κ⊓:

$A_2 = A_1 \cup \{ (\neg A \sqcup C)(\alpha), (\neg C \sqcup \exists R.D)(\alpha), (\neg D \sqcup \neg A)(\alpha), (\neg C \sqcup \forall R.A)(\alpha) \}$

- Εφαρμογη Κ⊔ στο $(\neg A \sqcup C)(\alpha)$

$A_3 = A_2 \cup \{ \neg A(\alpha) \}$
$A_4 = A_2 \cup \{ C(\alpha) \}$

Το A_3 περιεχει προφανη αντιφαση $(A(\alpha), \neg A(\alpha))$
εχουμε το A_4, εφαρμογη Κ⊔ στο $(\neg C \sqcup \exists R.D)(\alpha)$

$A_5 = A_4 \cup \{ \neg C(\alpha) \}$
$A_6 = A_4 \cup \{ \exists R.D(\alpha) \}$

Το A_5 περιεχει προφανη αντιφαση $(C(\alpha), \neg C(\alpha))$
εχουμε το A_6, εφαρμογη Κ∃ στο $\exists R.D(\alpha)$

$A_7 = A_6 \cup \{R(\alpha, x), D(x)\}$

Εφαρμογη του Κ∪ στο $(\neg D \cup \neg A)(\alpha)$

$A_8 = A_7 \cup \{\neg D(\alpha)\}$

$A_9 = A_7 \cup \{\neg A(\alpha)\}$

το A_9 περιεχει προφανη αντιφαση $\{A(\alpha), \neg A(\alpha)\}$

Εχουμε το A_8 εφαρμογη του Κ∪ στο $(\neg C \cup \forall R A)(\alpha)$

$A_{10} = A_8 \cup \{\neg C(\alpha)\}$

$A_{11} = A_8 \cup \{\forall R. A(\alpha)\}$

Το A_{10} περιεχει προφανη αντιφαση $(C(\alpha), \neg C(\alpha))$

εχουμε το A_{11}, εφαρμογη του Κ∀ στο $\forall R \circ A(\alpha)$

$A_{12} = A_{11} \cup \{A(x)\}$

Εφαρμογη εσωτερικευσης στο x:

$A_{13} = A_{12} \cup \{(\neg A \cup C) \cap (\neg C \cup \exists R.D) \cap (\neg D \cup \neg A) \cap (\neg C \cup \forall R A)(x)\}$

Εφαρμογη Κ∩ στο τελευταιο ισχυρισμο:

$A_{14} = A_{13} \cup \{(\neg A \cup C)(x), (\neg C \cup \exists R.D)(x), (\neg D \cup \neg A)(x),$
$(\neg C \cup \forall R.A)(x)\}$

Εφαρμογη του Κ∪ στο $(\neg D \cup \neg A)(x)$

$A_{15} = A_{14} \cup \{\neg D(x)\}$

$A_{16} = A_{14} \cup \{\neg A(x)\}$

το $D(x) \in A_7$ άρα το A_{15} περιέχει προφανή αντίφαση $(D(x), \neg D(x))$.
Το $A(x) \in A_{12}$ άρα το A_{16} περιέχει προφανή αντίφαση $(A(x), \neg A(x))$

→ Όλα τα ABox που παράχθηκαν από τον tableau περιέχουν αντίφαση, άρα κανένα δεν είναι clash-free

Άρα το A μβτ Σ <u>ΔΕΝ</u> είναι ικανοποιήσιμο!

<u>Σχόλιο:</u> Τυπικά ο tableau θα επεξεργαζόταν τις αντιφατικά ABoxes και θα τις απέρριπτε στο for-loop που θα έκανε έλεγχο για clash-free.
Ωστόσο για λόγους οικονομίας κάναμε τον έλεγχο για να μην γράφουμε πολλά, αφού θα απορρίφθουν στη συνέχεια!

Ερώτημα 3ο

(1.) Από την τριάδα ex:John ex:studiesAt ex:NTUA
μέσω RDF έχουμε ότι ex:studiesAt rdf:type rdf:Property
που ισοδύναμα γράφεται ως ex:studiesAt a rdf:Property

Απο την τριαδα ex:studiesAt rdfs:subPropertyOf ex:memberOf
και απο την τριαδα ex:John ex:studiesAt ex:NTUA
μεσω του rdfs7 εχουμε οτι:
ex:John ex:memberOf ex:NTUA (I)

Απο το (I) και την τριαδα ex:memberOf rdfs:domain ex:Person
μεσω του rdfs2 εχουμε οτι:
ex:John rdf:type ex:Person

(2.) Γενικα με την SPARQL μπορουμε να ρωτησουμε πολλα διαφορετικα ερωτηματα για τις δεδομενες τριαδες RDF ετσι ωστε να παρουμε το ζητουμενο αποτελεσμα.
Μια εκ των οποιων είναι:

```
SELECT ?x ?y
WHERE { {?x ex:worksAt ?y} UNION
        {?x ex:studiesAt ?y}
      }
```

που ουσιαστικα δινει την ενωση των ατομων (Person) που δουλευουν ή σπουδαζουν στο NTUA (χωρίς τη της βασης γνωσης)

Γενικοτερα θα είχαμε πολλες περισσοτερες δυνατοτητες για ερωτηματα, αλλα αυτες οι δυνατοτητες εξαρτουνται απο την βαση γνωση δηλαδη απο τις τριαδες RDF.
Δηλαδη τα queries θα μπορουσαν να ηταν πολυ πιο

συγκεκριμενα πανω στην γνωση μας, πχ να μην ειχαμε ως μεταβλητη το ακαδημαικο οργανισμο αλλα να αναφεραμε οτι ηταν το ex:NTUA. Το αποτελεσμα θα ηταν ακριβως ο ιδιος πινακας για αυτη την γνωση.

Το ερωτημα μας ειναι πιο γενικο απο αυτο αλλα θα μπορουσε να ηταν αναφορετικα γενικοτερο για μια αλλη γνωση οπου πχ ατομα θα μπορουσαν να δουλευαν και σε εταιριες και για οργανισμους.

Επισης η SPARQL δινει πολλα εργαλεια που θα μπορουσαν να μας βοηθησουν πχ τα φιλτρα, οπου να ελεγχαμε αν το 3ο ειναι το NTUA.

Επισης, θα μπορουσαμε πριν να κανουμε μια προεργασια στην γνωση τροποποιοντας την καταλληλα ετσι ωστε μετα να κανουμε το καταλληλο ερωτημα, πραγμα που γενικα δεν προτεινεται.

Αρα καταληγουμε στο συμπερασμα οτι η SPARQL ειναι πολυ εκφραστικη, αλλα και οτι η γνωση μας δινει πολλες δυνατοτητες για ερωτηματα πχ μεσα του ex:memberOf θα μπορουσαμε να πετυχαναμε τον στοχο μας.

Αρα ενα ερωτημα SPARQL απαιτει καλη γνωση των τριαδων RDF

Θέμα 4ο

α) Ακολουθεί ένα-προς-ένα αντιστοιχιση με τις τριαδες RDF που δινονται:

DisjointClasses(ex:Place ex:Person)
ObjectPropertyDomain(ex:livesIn ex:Person)
ObjectPropertyRange(ex:livesIn ex:Place)
ObjectPropertyAssertion(ex:livesIn ex:John ex:Athens)
ClassAssertion(_:b1 ex:Athens)
EquivalentClasses(_:b1 ObjectComplementOf(ex:Person))

β) Αρχικα θα κανουμε εναν μετασχηματισμο στα string για απλοτητα
Θεωρουμε ότι: $ex{:}Person \equiv A$, $ex{:}Place \equiv B$, $\_{:}b1 \equiv C$
$ex{:}livesIn \equiv R$
$ex{:}John \equiv a$ $ex{:}Athens \equiv b$
Οποτε χρησιμοποιόντας αυτες τις ονομασιες και χρησιμοποιόντας του πινακα των διαφανειων οι προηγουμενες τριάδες γινονται:

$$B \sqsubseteq \neg A$$
$$\exists R.\top \sqsubseteq A$$
$$\top \sqsubseteq \forall R.B$$
$$R(a,b)$$
$$\left.\begin{array}{l} C(b) \\ C \equiv \neg A \end{array}\right\} \rightarrow \neg A(b)$$

Έτσι, για να αποδειξει ενας OWL reasoner τα $C(b)$ και $C \equiv \neg A$ αρκει να αποδειξει το $\neg A(b)$.

Αρα, τωρα αρκει χρησιμοποιωντας τον αλγοριθμο tableau να αποδειξουμε οτι το $\mathcal{A} = \{ R(a,b), \neg A(b) \}$ είναι ικανοποιήσιμο μετ. $\mathcal{T} = \{ B \sqsubseteq \neg A, \exists R.\top \sqsubseteq A, \top \sqsubseteq \forall R.B \}$.

Επομένως έχουμε να εφαρμόσουμε τον αλγόριθμο tableau με δεδομένο ABox και TBox.
Έτσι, πρέπει να εφαρμόσουμε αρχικά TBox elimination
Έτσι πρέπει να προσθέσουμε για καθε άτομο του κόσμου τον ισχυρισμό
$(\neg B \sqcup \neg A) \sqcap (\neg(\exists R.\top) \sqcup A) \sqcap (\neg\top \sqcup \forall R.B) \equiv$
$(\neg B \sqcup \neg A) \sqcap (\forall R.\bot \sqcup A) \sqcap (\bot \sqcup \forall R.B)$
και η δομή tableau θα έχει αρχικά το δεδομένο ABox
και εφαρμογή του προηγούμενου κανόνα για τα άτομα α και b.

$S_0 = \{ \{ R(\alpha,b), \neg A(b), ((\neg B \sqcup \neg A) \sqcap (\forall R.\bot \sqcup A) \sqcap (\bot \sqcup \forall R.B)(\alpha),$
$((\neg B \sqcup \neg A) \sqcap (\forall R.\bot \sqcup A) \sqcap (\bot \sqcup \forall R.B)(b) \} \}$

Εφαρμόζοντας τον Κανόνα-$\sqcap$:

$S_1 = \{ \underbrace{\{ R(\alpha,b), \neg A(b), ((\neg B \sqcup \neg A) \sqcap (\forall R.\bot \sqcup A) \sqcap (\bot \sqcup \forall R.B))(\alpha),$
$((\neg B \sqcup \neg A) \sqcap (\forall R.\bot \sqcup A) \sqcap (\bot \sqcup \forall R.B))(b),$
$(\neg B \sqcup \neg A)(\alpha), (\forall R.\bot \sqcup A)(\alpha), (\bot \sqcup \forall R.B)(\alpha),$
$(\neg B \sqcup \neg A)(b), (\forall R.\bot \sqcup A)(b), (\bot \sqcup \forall R.B)(b) \}}_{A_1} \}$

Εφαρμόζοντας τον κανόνα-$\sqcup$: στο $(\neg B \sqcup \neg A)(\alpha)$

$S_2 = \{ \underbrace{A_1 \cup \{\neg B(\alpha)\}}_{A_3}, \underbrace{A_1 \cup \{\neg A(\alpha)\}}_{A_4} \}$

Εφαρμόζοντας τον K-$\sqcup$ στο $(\forall R.\bot \sqcup A)(\alpha)$:

$A_5 = A_3 \cup \{\forall R.\bot(\alpha)\}$
$A_6 = A_3 \cup \{A(\alpha)\}$ με $S_3 = \{A_5, A_6, A_7, A_8\}$
$A_7 = A_4 \cup \{\forall R.\bot(\alpha)\}$
$A_8 = A_4 \cup \{A(\alpha)\}$

Σημ: Το A_8 έχει προφανή αντίφαση $(A(\alpha), \neg A(\alpha))$

• Εφαρμογη ΚU στο $(\bot \sqcup \forall R.B)(\alpha)$:

$A_9 = A_5 \cup \{\bot(\alpha)\}$
$A_{10} = A_5 \cup \{\forall R.B(\alpha)\}$
$A_{11} = A_6 \cup \{\bot(\alpha)\}$
$A_{12} = A_6 \cup \{\forall R B(\alpha)\}$
$A_{13} = A_7 \cup \{\bot(\alpha)\}$
$A_{14} = A_7 \cup \{\forall R.B(\alpha)\}$
Τα A_9, A_{11}, A_{13} εχουν προφανης αντιφαση $(\bot(\alpha))$
εχουμε τα A_{10}, A_{12}, A_{14}

• Εφαρμογη Κ$\forall$ στο $\forall R\bot(\alpha)$ (Το $R(\alpha, \beta)$ υπαρχει στο αρχικο):

$A_{15} = A_{10} \cup \{\bot(\beta)\}$
$A_{16} = A_{14} \cup \{\bot(\beta)\}$
Τα A_{15}, A_{16} εχουν προφανης αντιφαση $(\bot(\beta))$

• Εφαρμογη Κ$\forall$ στο $\forall R B(\alpha)$:
$A_{17} = A_{22} \cup \{B(\beta)\}$

• Εφαρμογη Κ$\sqcap$ στο A_{17}:
$A_{18} = A_{17} \cup \{(\neg B \sqcap \neg A)(\beta), (\forall R\bot \sqcup A)(\beta), (\bot \sqcup \forall R.B)(\beta)\}$
• Εφαρμογη Κ$\sqcup$ στο $(\neg B \sqcap \neg A)(\beta)$:

$A_{19} = A_{18} \cup \{\neg B(\beta)\}$
$A_{20} = A_{19} \cup \{\neg A(\beta)\}$
Το A_{20} περιεχει προφανης αντιφαση $(B(\beta), \neg B(\beta))$

Εφαρμογη Κ$\sqcup$ στο $(\forall R.\bot \sqcup A)(\beta)$

$A_{21} = A_{20} \cup \{\forall R\bot(\beta)\}$
$A_{22} = A_{20} \cup \{A(\beta)\}$.

Το A_{22} εχει προφανης αντιφαση $(A(\theta), \neg A(\theta))$

- Εφαρμογη $K_\sqcup$ στο $(\perp \sqcup \forall R.B)(\theta)$

$A_{23} = A_{21} \cup \{\perp(\theta)\}$

$A_{24} = A_{22} \cup \{\forall R.B(\theta)\}$

Το A_{23} εχει προφανη αντιφαση $\perp(\theta)$

→ Και εδω τερματιζει ο αλγοριθμος Tableau επειδη δεν μπορουμε να εφαρμοσουμε αλλους κανονες.

Το A_{24} ειναι πληρες και clash-free

Αρα το A μαζι με το T που ειχαμε στην αρχη ή αλλιως η γνωση $K = \langle A, T \rangle$ ειναι ικανοποιησιμη αρα ο owl reasoner μπορει να αποφανθει οτι οι δυο τελευταιες τριαδες ειναι λογικα συμπερασμα

Σχολιο: Τυπικα ο Tableau θα επεξεργαζοταν τις αντιφατικες ABoxes και θα τις απεριπτει στο for loop που θα εκανε ελεγχο για clash-free. Ωστοσο εδω για λογους οικονομιας κανουμε τον ελεγχο για να μην γραφουμε παραπανω, αφου εξαλλου θα απορριφθουν στην συνεχεια.