Hedera Matúš - report RP

Plán na zimý semester:
Zoznámiť sa s už existujúcimi časťami projektu "Datalog ->Ra". Navrhnúť rozhranie medzi Datalogom a top-down výpočtom rekurzívnych dotazov v relačnej algebre.

Top-down výpočet:
Ak výpočet dotazu prebieha smerom "od dotazu nadol", eliminuje sa týmto postupom použitie pravidiel, ktoré sú z pohľadu výpočtu tohoto dotazu redundantné. Naivná evaluácia by pri výpočte použila aj tieto pravidlá.

Priebeh výpočtu:
Zavediem označenie pre pravidlá a dotazy *horný index*, ktoré musí vždy spĺňať nasldovné :
Ak hlava pravidla je označená *horným indexom* $i \in \mathbb{N}$, potom každý predikát v tele tohoto pravidla je označený *horným indexom* i, ak je v pozitívnom kontexte, inak (ak je v negatívnom kontexte) je označený *horným indexom* $i - 1$. Príklad:

$\mathbf{p^i(X) \leftarrow a^i(X), b^i(X), \neg c^{i-1}(X), \neg d^{i-1}(X), \neg e^{i-1}(X).}$

Pre ľubovoľné pravidlo q definujem $q^0 = \emptyset$
Výpočet ľubovoľného dotazu q začne výpočtom q^i pre $i = 1$. Ďalej sa v cykle opakujú nasledovné 3 kroky:

1.) inkrementuje sa i o 1
2.) vypočíta sa q^i
3.) ak $i \geq 2$, porovná sa q^i a q^{i-2}

V projekte "Datalog ->Ra" zodpovedajú pravidlám objekty *Operator*, realizujúce iteratívny výpočet, preto nemožno materializovať reláciu pre q^{i-2} v bode 3). Porovnanie relácií q^i a q^{i-2} teda znamená opätovný výpočet q^{i-2}.

Ukončenie výpočtu:
Ukončenie top-down výpočtu dotazu q nastane v momente, keď prvý krát $q^i = q^{i-2}$. Porovnávanie *horných indexov* spôsobom "ob 1 dozadu" zaručí, že výpočet zastane aj pre nestratifikovateľné dotazy. Ak je prvok vo všetkých 3 posledných iteráciach *horného indexu*, prehlásime ho za true. Ak nie je v žiadnej, prehlásime ho za false. Inak (ak alternuje) prehlásime ho za unknown.

Avšak takto popísaný spôsob ukončenia funguje len pre dotazy neobsahujúce v tele iné IDB predikáty (takéto predikáty sa už zrejme nebudú pre vyššie čísla *horných indexov* "meniť", rovnako ako hlava.)
Uvažujem príklad:

$\mathbf{r(a).}$
$\mathbf{q1(X) \leftarrow r(X)}$
$\mathbf{q2(X) \leftarrow r(X), \neg q1(X).}$
$\mathbf{q3(X) \leftarrow r(X), \neg q2(X).}$
$\mathbf{q3(X) =?}$

$\mathbf{q1^0 = \emptyset}$	$\mathbf{q1^1 = \{a\}}$	$\mathbf{q1^2 = \{a\}}$	$\mathbf{q1^3 = \{a\}}$	$\mathbf{q1^4 = \{a\}}$	$\mathbf{q1^5 = \{a\}}$
$\mathbf{q2^0 = \emptyset}$	$\mathbf{q2^1 = \{a\}}$	$\mathbf{q2^2 = \emptyset}$	$\mathbf{q2^3 = \emptyset}$	$\mathbf{q2^4 = \emptyset}$	$\mathbf{q2^5 = \emptyset}$
$\mathbf{q3^0 = \emptyset}$	$\mathbf{q3^1 = \{a\}}$	$\mathbf{q3^2 = \emptyset}$	$\mathbf{q3^3 = \{a\}}$	$\mathbf{q3^4 = \{a\}}$	$\mathbf{q3^5 = \{a\}}$

Popísaný algoritmus by zastavil už pri porovnaní $q3^0$ s $q3^2$ a a by prehlásil za unknown. Lenže od $q3^3$ a prestáva alternovať, preto správny verdikt pre a je true. Matematickou indukciou by som ľahko dokázal (napr. rozšírením myšlienky z predošlého príkladu), že pre ľubovoľný *horný index* i možno zostrojiť dotaz, pre ktorý nejaký prvok prestane alternovať až pre nejaký horný index väčší od i. Preto vo všeobecnosti nestačí o ukončení rozhodovať iba izolovane z pohľadu práve počítaného dotazu.

Rekurzívne definujem *Ukončiteľné* pravidlá. *Ukončiteľné* pravidlá budú také, ktoré neobsahujú v tele žiadne IDB predikáty, rôzne od hlavy pravidla, a sú označené najmenším *horným indexom* i spĺňajucím podmienku $q^i = q^{i-2}$ (teda použije sa vyššie popísaný spôsob ukončenia). *Ukončiteľné* pravidlá budú tiež také, v ktorých tele sú všetky predikáty *Ukončiteľné*, a sú označené najmenším *horným indexom* i spĺňajucím podmienku $q^i = q^{i-2}$.

Potom vo všeobecnosti sa výpočet dotazu ukončí v momente, keď je *Ukončiteľný*. V praxi to znamená nutnosť kontrolovať, či už sa všetky predikáty, od ktorých počítaný dotaz závisí, prestali "meniť". V tom prípade sú *Ukončiteľné* a výpočet možno ukončiť.

Výpočet pozitívnej časti tela pravidla:
Zavediem označenie pre pravidlá a dotazy *dolný index*, ktoré bude značiť číslo iterácie výpočtu naivnej evaluácie pre pozitívnu časť pravidla. Potom pre ľubovoľný dotaz q a $i \in N$, vypočet q^i začne výpočtom q^i_j pre $j = 1$. Ďalej sa v cykle opakujú nasledovné 3 kroky:

1.) inkrementuje sa j o 1
2.) vypočíta sa q^i_j
3.) porovná sa q^i_j a q^i_{j-1}

Výpočet pozitívnej časti q^i zastaví v momente, keď prvý krát $q^i_j = q^i_{j-1}$. V tomto prípade skutočne stačí porovnávať 2 po sebe nasledujúce iterácie, keďže vďaka pozitívnemu kontextu môžu zrejme nastať len 2 možnosti. Buď $q^i_j \supset q^i_{j-1}$ alebo $q^i_j = q^i_{j-1}$.