Riešenie konfliktov vo vyvrátiteľnom a dynamickom logickom programe

Jozef Frtús, Ján Šefránek

Katedra aplikovanej informatiky, FMFI, Univerzita Komenského Bratislava,
Mlynská dolina, 842 48, Bratislava
frtus@fmph.uniba.sk, sefranek@ii.fmph.uniba.sk

Abstrakt [1] Formalizmy založené na vyvrátiteľnej argumentácií sú v centre pozornosti nemonotónneho usudzovania a reprezentácií znalostí. Ich zaujímavou charakterizáciou je práve samotný argumentačný spôsob vyhodnocovania konfliktov. Na druhej strane, argumentačným formalizmom chýba modelovo-teoretická sémantika, preto ich porovnanie s formalizmami, ktore takúto sémantiku majú a tiež riešia konflikty, môže byť zaujímavé. Tento príspevok študuje vzťah medzi konkrétnymi formalizmami vyvrátiteľného logického programu (*defeasible logic program*) a dynamickým logickým programom (*update sequence*). Uvažujeme dva rôzne spôsoby transformácie vyvrátiteľného programu na dynamický program. Identifikovali sme niektoré podobné črty a niektoré odlišnosti oboch formalizmov. Najprv študujeme ako formalizmy riešia tzv. *nepriamy konflikt*. Ukážeme, že teoreticko-modelová sémantika nie je schopná tieto typy konfliktov riešiť. Dokážeme za akých predpokladov je množina potvrdených literálov vyvrátiteľného programu po transformácii stabilným modelom dynamického programu.

Kľúčové slová: riešenie konfliktov, argumentačný formalizmus, dynamický logický program

1 Úvod

V priebehu posledného desaťročia sa reprezentácia vyvíjajúcich sa znalostí stala neodmysliteľnou súčasťou teoretických základov umelej inteligencie. Základným problémom v tejto oblasti je riešenie konfliktov, to je vysporiadanie sa s protichodnými tvrdeniami. Bolo navrhnutých niekoľko rôznych formalizmov: argumentačné systémy, nemotónne logiky, dynamické logické programovanie. Každý z nich vníma problémy svojou optikou. Výsledky z jednej oblasti môžu byť inšpiráciou pre iné. Snahy o jednotný zovšeobecňujúci pohľad a porovnanie sú však ojedinelé.

Cieľom tohto príspevku je porovnať riešenie konfliktov v dvoch konkrétnych formalizmoch. Zameriame sa na argumentačný formalizmus reprezentovaný vyvrátiteľným logickým programom [4] na jednej strane a dynamické logické prog-

[1] Tento príspevok bol podporený grantom VEGA 1/0688/10

© Jelínek J., Jiroušek R. (Eds.): Znalosti 2011, pp. 54–65, ISBN 978-80-248-2369-0.
VŠB–Technická univerzita Ostrava, Fakulta elektrotechniky a informatiky, 2011.

ramy na strane druhej [3]. Pozornosť sústredime na spôsob riešenia tzv. *nepriameho konfliktu* a navrhneme spôsob ako ho odstrániť.

V druhej a tretej sekcii zhrnieme základné definície potrebné pre oboznámenie sa s dynamickým a vyvrátiteľným logickým programom. V štvrtej sekcii sa venujeme problematike riešenia konfliktov: po definícii tzv. nepriameho konfliktu nasleduje navrhnutie transformácie vyvrátiteľného pravidla a samotné porovnanie sémantiky formalizmov.

2 Dynamický logický program

2.1 Rozšírený logický program

Nech $\mathcal{L}$ je množina atómov. Objektívny literál je atóm L, alebo jeho explicitná negácia $\neg L$. Pre objektívny literál L platí $\neg\neg L = L$. Množinu všetkých objektívnych literálov označíme Obj. Literál je objektívny literál L alebo jeho defaultová negácia $not\ L$.

Pravidlo r je usporiadaná dvojica $H(r) \leftarrow B(r)$, kde hlava pravidla $H(r) = L_0$ je objektívny literál a telo pravidla $B(r) = \{L_1, \dots, L_m, notL_{m+1}, \dots, notL_n\}$ je konečná množina literálov. Ak $B(r) = \emptyset$, potom pravidlo r nazývame fakt a miesto $A \leftarrow$, píšeme len A. Ďalej, $B(r)^+, B(r)^-$ sú množiny objektívnych literálov, $B(r)^+ = \{L_1, \dots, L_m\}$, $B(r)^- = \{L_{m+1}, \dots, L_n\}$, pravidlo r^+ označuje $H(r) \leftarrow B(r)^+$. Rozšírený logický program (ďalej len program) je konečná množina pravidiel.

Interpretácia je množina objektívnych literálov, ktorá súčasne neobsahuje komplementárny pár A a $\neg A$. Literál L je v interpretácii I pravdivý ($I \models L$), ak $L \in I$, inak je nepravdivý ($I \not\models L$). Telo pravidla je v interpretácii I pravdivé ($I \models B(r)$) ak, $\forall A \in B(r)^+,\ I \models A \wedge \forall L \in B(r)^-,\ I \not\models L$. Pravidlo r je v interpretácii I splnené ($I \models r$), ak $I \models B(r)$ implikuje $I \models H(r)$. Interpretácia I je modelom programu P, ak $\forall r \in P, I \models r$.

Nech P je program, S interpretácia, potom P^S redukcia je množina pravidiel $\{r^+ | r \in P,\ B(r)^- \cap S = \emptyset\}$. S je stabilný model (answer set) programu P, ak $least(P^S) = S$, kde $least(.)$ je štandardný operátor odvodenia pre definitné logické programy, ktorý „vypočíta" najmenší model.

2.2 Postupnosť aktualizácií

Postupnosť programov reprezentuje časový vývin dynamicky sa meniacich znalostí o svete. Každý program v postupnosti reprezentuje aktualizáciu predchádzajúcich. Novšia informácia teda môže spôsobiť falzifikáciu predošlých. Aktualizácie logických programov (*update programs*) [3] uvádzame v jazyku rozšírených logických programov, čo uľahčí porovnávanie s vyvrátiteľným logickým programom, ktorý obsahuje explicitnú negáciu.

Pod postupnosťou aktualizácií $\boldsymbol{P}$ rozumieme postupnosť $(P_1, \dots, P_n)$ rozšírených logických programov. Ďalej, miesto $r \in P_i, 1 \leq i \leq n$ budeme písať $r \in \boldsymbol{P}$. Ak $\boldsymbol{P} = (P_1, \dots, P_n)$ je postupnosť aktualizácií, potom $|\boldsymbol{P}| = n$ značí počet programov v postupnosti.

Definícia 1. ([3]) *Nech* $\boldsymbol{P} = (P_1, \ldots, P_n)$ *je postupnosť aktualizácií nad* $\mathcal{L}$ *a* S *je množina literálov. Potom* S *je stabilný model* $\boldsymbol{P}$ *práve vtedy keď* S *je minimálny model* $(\bigcup \boldsymbol{P} \setminus Reject(\boldsymbol{P}, S))^S$*, kde:*

- $\bigcup \boldsymbol{P} = \bigcup_{i=1}^{n} P_i$
- $Reject(\boldsymbol{P}, S) = \bigcup_{i=1}^{n} Rej_i(\boldsymbol{P}, S)$
- $Rej_n(\boldsymbol{P}, S) = \emptyset$
- $Rej_i(\boldsymbol{P}, S) = \{r \in P_i | \exists q \in P_j \setminus Rej_j(\boldsymbol{P}, S), j \in \{i+1, \ldots, n\}, H(r) = \neg H(q) \wedge S \models B(r) \cup B(q)\}, i \in \{1, \ldots, n-1\}$

3 Vyvrátiteľný logický program

Idea vyvrátiteľného logického programu vznikla na pomedzí výskumu konvergujúcich oblastí nemonotónneho usudzovania, logického programovania a argumentačných formalizmov. Vyvrátiteľný logický program umožňuje reprezentovať poznatky vo forme *vyvrátiteľných* (t.j. spochybniteľných) pravidiel. Z týchto pravidiel ďalej vytvárame argumentačné štuktúry - minimálne množiny pravidiel potrebné k odvodeniu nejakej hypotézy. Konflikty (protichodné informácie) sa riešia prostredníctvom vyhodnocovania argumentov v štýle argumentácie: „Pravdu má ten, kto má posledné slovo“. Inferencia nad vyvrátiteľným programom je realizovaná procedúrou, ktorá počíta zaručene pravdivé (*warranted*) hypotézy. Intuitívne, hypotézu prijmeme za pravdivú, ak ju „ubránime“ pred všetkými kontraargumentami.

Vyvrátiteľný logický program, obsahuje dva druhy pravidiel: striktné, a vyvrátiteľné. Vyvrátiteľné pravidlá reprezentujú neisté, spochybniteľné znalosti: pravdivosť tela pravidla nezaručuje pravdivosť dôsledku pravidla. Vyvrátiteľné pravidlo, ktoré značíme $A \prec B$ sa číta ako: „ak platí B, zvyčajne platí aj A“.

Vo vyvrátiteľnom logickom programe štandardne uvažujeme len objektívne literály. Preto, ak to nebude viesť k nejednoznačnosti, keď budeme hovoriť o literáli, budeme mať na mysli objektívny literál. Vyvrátiteľný logický program $\mathcal{P}$ je usporiadaná dvojica disjunktných množín (Π, Δ), kde Π je konečná konzistentná [2] množina striktných pravidiel a Δ konečná množina vyvrátiteľných pravidiel. Striktné pravidlo r je usporiadaná dvojica $H(r) \leftarrow B(r)$, kde $H(r) = L_0$ je literál a $B(r) = \{L_1, \ldots, L_n\}$ je konečná množina literálov. Ak $B(r) = \emptyset$, potom pravidlo r nazývame fakt. Vyvrátiteľné pravidlo r je usporiadaná dvojica $H(r) \prec B(r)$, kde $H(r) = L_0$ je literál a $B(r) = \{L_1, \ldots, L_n\}$ je konečná neprázdna množina literálov. Pod $H(\Gamma), \Gamma \subseteq \Pi \cup \Delta$ ďalej rozumieme $\{H(r) | r \in \Gamma\}$, kde $H(r)$ je hlava pravidla r.

Ďalej pod množinou $\Gamma \subseteq \mathcal{P}$ rozumieme $\Gamma \subseteq \Pi \cup \Delta$.

Definícia 2. *(Vyvrátiteľné odvodenie) Nech* $\mathcal{P}$ *je vyvrátiteľný program,* L *literál a* $\Gamma \subseteq \mathcal{P}$ *množina pravidiel. Vyvrátiteľné odvodenie* L *z* Γ ($\Gamma \mathrel{|\!\!\sim} L$) *je postupnosť literálov* $\rho : L_1, \ldots, L_n = L$*, pričom pre každý* $L_i, i \in \{1, \ldots, n\}$ *platí:*
$(\exists r \in \Gamma, H(r) = L_i) \wedge (\forall B_j \in B(r), \exists k \in \{1, \ldots, i-1\}, B_j = L_k)$

[2] pre $\forall L \in Obj$, $L \in Cn(\Pi)$ implikuje $\neg L \notin Cn(\Pi)$, kde Cn je štandardný operátor odvodenia

Odvodenie literálu L z množiny Γ nazývame striktné ($\Gamma \vdash L$), ak $\Gamma \subseteq \Pi$.

Definícia 3. *(Nekonzistentá množina pravidiel) Množina pravidiel $\Gamma \subseteq \mathcal{P}$ je nekonzistentná ak existuje literál L taký, že $\Gamma \mid\!\sim L, \Gamma \mid\!\sim \neg L$. Množina, ktorá nie je nekonzistentá, je konzistentná.*

Evidentne, ak je $\Gamma \subseteq \mathcal{P}$ nekonzistená, potom $\Gamma \cap \Delta \neq \emptyset$. Množina striktných pravidiel Π je totiž podľa definície konzistentná.

Definícia 4. *(Argumentačná štruktúra) Nech $\mathcal{P} = (\Pi, \Delta)$ je vyvrátiteľný program, h je literál a $A \subseteq \Delta$ je množina vyvrátiteľných pravidiel. Argumentačná štruktúra (ďalej len argument) je dvojica $\mathcal{A} = (A, h)$, pre ktorú platí:*

1. *$\Pi \cup A \mid\!\sim h$*
2. *$\Pi \cup A$ je konzistentná*
3. *$\nexists A' \subset A$, ktorá by spĺňala podmienky (1), (2)*

Množinu všetkých argumentov, ktoré vieme z vyvrátiteľného programu $\mathcal{P}$ extrahovať označíme $Args_{\mathcal{P}}$.

Príklad 1. Nech $\mathcal{P} = (\Pi, \Delta)$ je vyvrátiteľný program

$$\Pi = \left\{ \begin{array}{l} a.\ e. \\ \neg\, b \leftarrow c. \\ d \leftarrow a. \end{array} \right\} \Delta = \left\{ \begin{array}{l} r_1 : c \prec a. \\ r_2 : d \prec e. \\ r_3 : \neg c \prec e. \end{array} \right\}$$

Množina všetkých argumentov, ktoré vieme z $\mathcal{P}$ extrahovať je $Args_{\mathcal{P}} = \{(\emptyset, a), (\emptyset, e)$, $(\emptyset, d)$, $(\{c \prec a\}, c), (\{c \prec a\}, \neg b)$, $(\{\neg c \prec e\}, \neg c)\}$.

Príklad okrem iného demonštruje, že ak $\mathcal{P} = (\Pi, \Delta)$ je vyvrátiteľný program, $\mathcal{A} = (\emptyset, h)$ argument, potom $\Pi \vdash h$.

Nech $\mathcal{A} = (A, h)$ a $\mathcal{B} = (B, i)$ sú argumenty. Hovoríme, že $\mathcal{B}$ je podargumentom $\mathcal{A}$ ak $B \subseteq A$.

Definícia 5. *(Nesúhlasné literály) Nech (Π, Δ) je vyvrátiteľný program. Literály h, q sú nesúhlasné, ak $\Pi \cup \{h, q\}$ je nekonzistentá množina.*

Vo vyvrátiteľnom programe pri rozhodovaní medzi konfliktnými hypotézami nie je nutné explicitne zadávať reláciu preferencie na pravidlách. Výber platnej hypotézy je docielený procedurálnym vyhodnocovaním argumentov. Avšak, dodatočné znalosti (aktuálnosť, vyššia priorita) o vyvrátiteľných pravidlách vo forme ich vzájomných preferencií môže zvýšiť relevantnosť reprezentovaných znalostí. V našej práci budeme v nasledujúcom predpokladať nejakú abstraktnú reláciu preferencie nad vyvrátiteľnými pravidlami $\prec$. Intuitívne, pravidlo r_2 je preferovanejšie ako r_1, ak $(r_1, r_2) \in \prec$.

Nech $\prec$ je relácia preferencie nad vyvrátiteľnými pravidlami a $\mathcal{A} = (A, h)$, $\mathcal{B} = (B, i)$ sú argumenty. Argument $\mathcal{B}$ je preferovaný nad argumentom $\mathcal{A}$, ($\mathcal{A} \prec \mathcal{B}$), ak $(\exists r_1 \in A, r_2 \in B, r_1 \prec r_2) \wedge (\nexists r_3 \in A, r_4 \in B, r_4 \prec r_3)$

Definícia 6. *(Silný protiargument) Nech $\mathcal{A} = (A, h)$ a $\mathcal{B} = (B, l)$ sú argumenty. Hovoríme, že argument $\mathcal{A}$ je silným protiargumentom $\mathcal{B}$ zacielený na literál i vzhľadom na reláciu $\prec$, ak $\mathcal{C} = (C, i)$ je podargumentom $\mathcal{B}$, literály h a i sú nesúhlasné a argument $\mathcal{A}$ je preferovaný nad argumentom $\mathcal{C}$, ($\mathcal{C} \prec \mathcal{A}$).*

Definícia 7. *(Blokujúci protiargument) Nech* $\mathcal{A} = (A, h)$ *a* $\mathcal{B} = (B, l)$ *sú argumenty. Hovoríme, že argument* $\mathcal{A}$ *je blokujúcim protiargumentom* $\mathcal{B}$ *zacielený na literál i vzhľadom na reláciu* $\prec$*, ak* (C, i) *je podargumentom* $\mathcal{B}$*, literály h a i sú nesúhlasné a* $(\mathcal{C}, \mathcal{A}) \notin \prec$*, ani* $(\mathcal{A}, \mathcal{C}) \notin \prec$*.*

Definícia 8. *(Protiargument) Nech* $\mathcal{A}$ *a* $\mathcal{B}$ *sú argumenty.* $\mathcal{A}$ *je protiargumentom* $\mathcal{B}$*, ak* $\mathcal{A}$ *je silným alebo blokujúcim protiargumentom* $\mathcal{B}$*.*

Definícia 9. *(Relácia atakovania) Pod reláciou atakovania* $\mathcal{R}$ *rozumieme množinu* $\mathcal{R} = \{(\mathcal{A}, \mathcal{B}) | \ \mathcal{A}$ *je protiargumentom* $\mathcal{B}\}$*.*

Tvrdenie 1. ([4]) *Pre argument* $\mathcal{A} = (\emptyset, h)$ *neexistuje protiargument.*

Definícia 10. *(Zhoda argumentov) Nech* $\mathcal{P} = (\Pi, \Delta)$ *je vyvrátiteľný program. Hovoríme, že množina argumentov* $\mathcal{A}_i = (A_i, h_i)$ *je v zhode, ak množina pravidiel* $\bigcup_i A_i \cup \Pi$ *je konzistentná.*

Protiargumenty identifikujú a poukazujú na tie miesta našich znalostí, na ktoré môže niekto namietať, nesúhlasiť s nimi. Protiargumenty sú však tiež argumentačné štruktúry a môžu byť teda tiež napadnuté iným protiargumentom. Argumentácia pripomína istú formu dialógu, kde vyslovený argument je protiargumentom tomu predchádzajúcemu. Nasledovnú definíciu *akceptovateľnej argumentačnej línie* môžeme teda považovať za istú formalizáciu takejto argumentácie.

Definícia 11. *(Akceptovateľná argumentačná línia) Konečnú postupnosť argumentov* $\Lambda = (\mathcal{A}_0, \ldots, \mathcal{A}_n)$ *nazveme akceptovateľnou argumentačnou líniou, ak:*

- $\forall \mathcal{A}_i \in \Lambda, i \in \{0, \ldots, n-1\}, (\mathcal{A}_{i+1}, \mathcal{A}_i) \in \mathcal{R}$
- *množina argumentov* $\bigcup \mathcal{A}_i, i$ *je párne číslo, podporujúcich argument* $\mathcal{A}_0$ *je v zhode.*
- *množina argumentov* $\bigcup \mathcal{A}_i, i$ *je nepárne číslo, atakujúcich argument* $\mathcal{A}_0$ *je v zhode.*
- $\forall \mathcal{A}_i \in \Lambda \ \nexists \mathcal{A}_j \in \Lambda, \ i < j, \ \mathcal{A}_j$ *je podargumentom* $\mathcal{A}_i$*.*
- $\forall \mathcal{A}_i \in \Lambda$*, ak* $\mathcal{A}_i$ *je blokujúci protiargument* $\mathcal{A}_{i-1}$*, potom* $\mathcal{A}_{i+1} \in \Lambda$ *implikuje* $\mathcal{A}_{i+1}$ *je silný protiargument* $\mathcal{A}_i$*.*

Aby sme nejakú hypotézu h mohli považovať za potvrdenú, musí existovať argument $(\mathcal{A}, h)$, ktorý sa ubráni voči všetkým protiargumentom. Tento proces vyhodnocovania je rekurzívny a zodpovedá prezeraniu stromu, kde priamy nasledovník je protiargumentom pre daný argument.

Definícia 12. *(Dialektický strom) Nech* $\mathcal{A}_0$ *je argument. Dialektický strom pre argument* $\mathcal{A}_0$ *(značíme* $\mathcal{T}_{\mathcal{A}_0}$*) je definovaný nasledovne:*

- *Koreň stromu je označený* $\mathcal{A}_0$*.*
- *Nech vnútorný vrchol V je označený* $\mathcal{A}_n$*,* $(\mathcal{A}_0, \ldots, \mathcal{A}_n)$ *je postupnosť argumentov, ktorými sú označené vrcholy na ceste z koreňa stromu do vrchola V. Ďalej, nech* $\mathcal{B} = \{\mathcal{B}_1, \ldots, \mathcal{B}_k\}$ *je množina všetkých protiargumentov* $\mathcal{A}_n$*. Ak postupnosť* $\Lambda_i = (\mathcal{A}_0, \ldots, \mathcal{A}_n, \mathcal{B}_i), 1 \leq i \leq k$ *je akceptovateľná argumentačná*

línia, potom vrchol V má nasledovníka V_i s označením $\mathcal{B}_i$. Ak množina $\mathcal{B} = \emptyset$ alebo $\forall i, \Lambda_i$ nie je akceptovateľná argumentačná línia, potom vrchol V je list stromu.

Nasleduje samotná procedúra, ktorá vyhodnocuje dialektický strom. Po jej ukončení sú jednotlivé vrcholy stromu vyhodnotené ako „P“ (porazené), resp. „N“ (neporazené).

Procedúra 1 *(Vyhodnocovanie dialektického stromu) Nech $\mathcal{T}_{\mathcal{A}_0}$ je dialektický strom pre argument $\mathcal{A}_0$. Vyhodnotený strom značíme $\mathcal{T}^*_{\mathcal{A}_0}$, pričom vyhodnocovacia procedúra je definovaná nasledovne:*

1. *Všetky listy v strome $\mathcal{T}_{\mathcal{A}_0}$ sú vyhodnotené ako „N“.*
2. *Nech V je vnútorný vrchol v strome $\mathcal{T}_{\mathcal{A}_0}$. Potom V je vyhodnotený ako „N“, ak všetci jeho nasledovníci sú ohodnotení ako „P“. V je vyhodnotený ako „P“, ak existuje apoň jeden jeho nasledovník, ktorý je ohodnotený ako „N“.*

Definícia 13. *(Potvrdený literál) Nech $\mathcal{A} = (A, h)$ je argument a $\mathcal{T}^*_{\mathcal{A}}$ jeho vyhodnotený dialektický strom. Ak koreň $\mathcal{T}^*_{\mathcal{A}}$ je označený ako „N“, potom literál h nazývame potvrdený.*

Množinu všetkých potvrdených literálov vyvrátiteľného programu $\mathcal{P}$ označíme $War(\mathcal{P})$.

Tvrdenie 2. ([4]) *Nech $\mathcal{P}$ je vyvrátiteľný program, h je literál. Ak $\mathcal{P} \vdash h$, potom $h \in War(\mathcal{P})$.*

4 Riešenie konfliktov

4.1 Nepriamy konflikt

V tejto sekcii zavedieme niekoľko nových pojmov o vyvrátiteľnom logickom programe. Pozornosť ďalej sústredime na špeciálny prípad konfliktov, ktorý môže nastať a v našom porovnaní formalizmov zohráva dôležitú úlohu.

Argument $\mathcal{A} = (A, h)$ nazývame singulárny, ak $|A| = 1$.

Definícia 14. *(Aplikovateľné pravidlo) Hovoríme, že pravidlo $r \in \mathcal{P}$ je aplikovateľné vzhľadom na množinu pravidiel $\Gamma \subseteq \Pi \cup \Delta$, ak platí $\Gamma \mathrel{|\!\!\sim} H(r)$.*

Pozorovanie 1 *Nech $\mathcal{P} = (\Pi, \Delta)$ je vyvrátiteľný program, $F = Cn(\Pi)$ množina deduktívnych dôsledkov z faktov v Π. Ak $Args_{\mathcal{P}}$ je množina len singulárnych argumentov a všetky pravidlá v Δ sú aplikovateľné, potom $\forall q \in \Delta$, $B(q) \subseteq F$.*

Uvedomme si, že pre danú hypotézu môže existovať viac ako jeden argument. Potom je praktické, zaviesť pojem najlepšieho argumentu pre hypotézu.

Definícia 15. *(Najlepší argument pre hypotézu) Nech $\mathcal{P} = (\Pi, \Delta)$ je vyvrátiteľný program, $Args_{\mathcal{P}}$ je množina len singulárnych argumentov, $\prec$ lineárne usporiadanie na Δ. Hovoríme, že argument $\mathcal{A} = (A, h)$ je pre hypotézu h vzhľadom na $\prec$ najlepší, ak $\nexists \mathcal{B} = (B, h) \in Args_{\mathcal{P}}$, že $\mathcal{A} \prec \mathcal{B}$. Pre daný literál h značíme najlepší argument $maxArg(h)$.*

Lemma 1. *Nech $\mathcal{P}$ je vyvrátiteľný program. Ak $Args_{\mathcal{P}}$ je množina len singulárnych argumentov, $\prec$ lineárne usporiadanie na Δ, potom $\nexists a, b \in War(\mathcal{P})$ také, že a, b sú nesúhlasné literály.*

Dôkaz. Sporom: Nech také nesúhlasné literály $a, b \in War(\mathcal{P})$ existujú. Bez ujmy na obecnosti nech $maxArg(a) = \mathcal{A} = (A, a)$, $maxArg(b) = \mathcal{B} = (B, b)$ a $\mathcal{B} \prec \mathcal{A}$. Potom z nesúhlasnosti a, b vyplýva, že $(\mathcal{A}, \mathcal{B}) \in \mathcal{R}$. Keďže $b \in War(\mathcal{P})$, potom $\exists \mathcal{C} \in Args_{\mathcal{P}}$, $(\mathcal{C}, \mathcal{A}) \in \mathcal{R}$ a zároveň $\mathcal{T}^*_{\mathcal{C}}$ je označený ako „N“. Ďalej pre $a \in War(\mathcal{P})$ potom $\exists \mathcal{D} \in Args_{\mathcal{P}}$, $(\mathcal{D}, \mathcal{C}) \in \mathcal{R}$ a zároveň $\mathcal{T}^*_{\mathcal{D}}$ je označený ako „N“. Existencia takého argumentu $\mathcal{D}$ je však v rozpore s tým, že $b \notin War(\mathcal{P})$. □

Definícia 16. *(Nepriamy konflikt) Nech $\mathcal{P} = (\Pi, \Delta)$ je vyvrátiteľný program, $Args_{\mathcal{P}}$ je množina len singulárnych argumentov, $\prec$ lineárne usporiadanie na Δ. Hovoríme, že v $\mathcal{P}$ je nepriamy konflikt, ak $\exists \mathcal{B} = (B, h), \mathcal{C} = (C, k) \in Args_{\mathcal{P}}, (\mathcal{B}, \mathcal{C}) \in \mathcal{R}$ (teda h, k sú nesúhlasné literály) $\wedge\ h \neq \neg k$. O literáloch h, k potom hovoríme, že sú v nepriamom konflikte.*

Príklad 2. Nech $\mathcal{P} = (\Pi, \Delta)$ je vyvrátiteľný program

$$\Pi = \left\{ \begin{array}{l} a.\ b. \\ f \leftarrow c. \\ \neg f \leftarrow d. \end{array} \right\} \qquad \Delta = \left\{ \begin{array}{l} r_1 : d \mathrel{-\!\!\prec} a. \\ r_2 : c \mathrel{-\!\!\prec} b. \end{array} \right\}$$

Nech $\prec = \{(r_1, r_2)\}$ je usporiadanie na vyvrátiteľných pravidlách. Uvažujme argumenty $\mathcal{A} = (\{r_1\}, d)$, $\mathcal{B} = (\{r_2\}, c)$. Vidíme, že literály c, d sú v nepriamom konflikte ($\Pi \cup \{c, d\}$ je nekonzistentná množina, $c \neq \neg d$).

Lemma 2. *Nech $\mathcal{P} = (\Pi, \Delta)$ je vyvrátiteľný program, $Args_{\mathcal{P}}$ je množina len singulárnych argumentov, $\prec$ lineárne usporiadanie na Δ, v Π nie je nepriamy konflikt, h je literál a $maxArg(h) = \mathcal{A} = (A, h) \in Args_{\mathcal{P}}$. Potom $h \in War(\mathcal{P})$ práve vtedy keď $\nexists \mathcal{B} \in Args_{\mathcal{P}}, (\mathcal{B}, \mathcal{A}) \in \mathcal{R}$.*

Dôkaz.
- „$\Rightarrow$“ Sporom, nech $\exists \mathcal{B} = (B, l) \in Args_{\mathcal{P}}, (\mathcal{B}, \mathcal{A}) \in \mathcal{R}$. Z predpokladu, že v Π nie je nepriamy konflikt dostávame, že $l = \neg h$. Keďže $h \in War(\mathcal{P})$, potom $\forall \mathcal{B} \in Args_{\mathcal{P}} \exists \mathcal{C} = (C, h) \in Args_{\mathcal{P}}, (\mathcal{B}, \mathcal{A}) \in \mathcal{R}$ implikuje $(\mathcal{C}, \mathcal{B}) \in \mathcal{R}$. Na základe linearity $\prec$ potom platí $\mathcal{A} \prec \mathcal{B} \prec \mathcal{C}$. To je však spor s tým, že $maxArg(h) = \mathcal{A}$.
- „$\Leftarrow$“ triviálne splnené. □

4.2 Navrhnutie transformácie

V nasledujúcom chceme transformovať vyvrátiteľný logický program na postupnosť aktualizácií a následne porovnať množinu potvrdených hypotéz s množinou stabilných modelov postupnosti aktualizácií. Aby sme toto porovnanie mohli realizovať, je nutné sa vysporiadať s prekladom vyvrátiteľného pravidla na pravidlo rozšíreného logického programu.

Transformáciu celého vyvrátiteľného programu zadefinujeme najprv všeobecne a abstraktne.

Definícia 17. *(ξ-transformácia vyvrátiteľného programu) Nech $\mathcal{P} = (\Pi, \Delta)$ je vyvrátiteľný program, $\prec$ je lineárne usporiadanie na Δ, ξ nejaká transformácia vyvrátiteľných pravidiel na pravidlá rozšíreného logického programu, $Args_{\mathcal{P}}$ množina len singulárnych argumentov. $\mathcal{P}$ je ξ-transformovaný na postupnosť aktualizácií $(P_1, \ldots, P_n, P_{n+1})$, kde*

- $P_{n+1} = \Pi$
- $P_i = \xi(r), r \in \Delta, i \in \{1, \ldots, n\}, n = |\Delta|$

Každá ξ-transformácia zachováva reláciu usporiadania $\prec$ v zmysle

- $\forall r, s \in \Delta$, $r \prec s$ *práve vtedy keď* $\xi(r) \in P_i$, $\xi(s) \in P_j$, $i < j$

Ďalej pod $\xi(\mathcal{P})$ budeme rozumieť postupnosť aktualizácií $\boldsymbol{P} = (P_1, \ldots, P_n)$, ktorá vznikla ξ-transformáciou $\mathcal{P}$. Pod označením $\xi(\mathcal{P})_i$ myslíme $P_i \in \boldsymbol{P}$ a podobne $\bigcup \xi(\mathcal{P})$ značí $\bigcup \boldsymbol{P}$.

Prirodzeným kandidátom na preklad $-\!\!<$ je interpretácia vyvrátiteľnosti vo forme defaultovej negácie hlavy pravidla.

Definícia 18. *(π-transformácia vyvrátiteľných pravidiel) Nech $\mathcal{P} = (\Pi, \Delta)$ je vyvrátiteľný program. Funkcia π je ξ-transformácia, ktorá transformuje každé pravidlo z Δ v tvare $A -\!\!< A_1, \ldots, A_n$ na pravidlo typu $A \leftarrow A_1, \ldots, A_n, not\ \neg A$.*

Takto navrhnutú transformáciu vyvrátiteľného pravidla použil Brewka pri preklade vyvrátiteľnej teórie na rozšírený logický program s preferenciami a dobre fundovanou sémantikou [1]. Garcia, Simari ukázali, že množina potvrdených literálov vyvrátiteľného programu nie je dobre fundovaným modelom logického programu uvažovaného v [1]. V našom prístupe, je vzťah charakterizovaný tvrdením 3.

4.3 Sémantická charakterizácia formalizmov

Porovnanie riešenia konfliktov predstavuje jadro tohto príspevku. V nasledujúcej sekcii sa o to pokúsime prostredníctvom hľadania nejakého vzťahu medzi množinou potvrdených literálov vyvrátiteľného programu a množinou stabilných modelov postupnosti aktualizácií.

Lemma 3. *Nech $\mathcal{P} = (\Pi, \Delta)$ je vyvrátiteľný program, $\boldsymbol{P} = (P_1, \ldots, P_m)$ jeho ľubovoľná ξ-transformácia na postupnosť aktualizácií. Ak $\Pi \vdash h$, potom $\forall M \in SM(\boldsymbol{P}), h \in M$.*

Dôkaz. $\Pi \vdash h$ znamená, že existuje postupnosť literálov $L_1, \ldots, L_n = h$, kde pre $L_i, i \in \{1, \ldots, n\}$ platí, že $(\exists r \in \Pi, H(r) = L_i) \wedge (\forall B \in B(r), \exists k \in \{1, \ldots, i-1\}, B = L_k)$. Ďalej vieme, že Π je konzistentná množina a ξ-transformácia zaručuje, že $P_m = \Pi$. Na základe def. 1 a predchádzajúcich pozorovaní už potom priamo vyplýva, že $(\exists r \in P_m)(\forall M \in SM(\boldsymbol{P}))$, $(H(r) = h \wedge M \models B(r))$, a teda aj $h \in M$.$\square$

Definícia 19. *(Najpreferovanejšie pravidlo pre literál) Nech* $\boldsymbol{P}$ *je postupnosť aktualizácií. Pre daný literál h* $maxP(h) = max\{j \in \{1, \dots, |\boldsymbol{P}|\} | \exists q \in P_j, H(q) = h\}$ *vráti najväčie číslo programu (v rámci* $\boldsymbol{P}$*) v ktorom sa nachádza pravidlo s hlavou h. Pravidlo r nazývame najpreferovanjším pre literál h, ak* $r \in P_{maxP(h)}$*. Pre daný literál h, najpreferovanejšie pravidlo značíme* $maxR(h)$*. Ďalej pod označením,* $maxP_S(h) = max\{j \in \{1, \dots, |\boldsymbol{P}|\} | \exists q \in P_j, H(q) = h \wedge S \models B(q)\}$ *myslíme číslo programu v ktorom sa nachádza najpreferovanejšie pravidlo r (pre literál h) s pravdivým telom vzhľadom na interpretáciu S. Analogicky* $maxR_S(h)$*.*

Definícia 20. *(Preferovaný model) Nech* $\boldsymbol{P} = (P_1, \dots, P_n)$ *je postupnosť aktualizácií vytvorená* ξ*-transformáciou vyvrátiteľného programu* $\mathcal{P}$*, ktorý obsahuje len singulárne argumenty. Ďalej, nech M je model* $\boldsymbol{P}$ *a* $F = Cn(P_n)$ *je množina deduktívnych dôsledkov z faktov v programe* P_n*. Hovoríme, že M je preferovaný model voči S (značíme* $S \ll M$*), ak platí* $\exists L \in M, \neg L \in S \wedge maxP_M(A) > max(maxP_S(\neg A), maxP_S(B))$*, kde A, B sú literály*[3]*:*

$$A : \begin{cases} B(maxR_M(L))^+ \setminus F & ak\ B(maxR_M(L))^+ \setminus F \neq \emptyset \\ L & v\ opačnom\ prípade \end{cases}$$

$$B : \begin{cases} B(maxR_S(\neg L))^+ \setminus F & ak\ B(maxR_S(\neg L))^+ \setminus F \neq \emptyset \\ \neg L & v\ opačnom\ prípade \end{cases}$$

Hovoríme, že M je preferovaný model programu $\boldsymbol{P}$*, ak neexistuje model* $N \neq M$*, že* $M \ll N$*.*

Príklad 3. Na vyvrátiteľnom programe $\mathcal{P}$ z príkladu 1 demonštrujeme π-transformác a pojem prefererovaného modelu. $\mathcal{P}$ je π-transformovaný na $(P_1, \dots, P_4)$, kde:

$$P_4 = \Pi = \{a.\ e.\ \neg\ b \leftarrow c.\ d \leftarrow a.\}$$
$$P_3 = \pi(\{r_3\}) = \{\neg c \leftarrow e, not\ c.\}$$
$$P_2 = \pi(\{r_2\}) = \{d \leftarrow e, not\ \neg d.\}$$
$$P_1 = \pi(\{r_1\}) = \{c \leftarrow a, not\ \neg c.\}$$

Uvažujme reláciu $\prec = \{(r_1, r_2), (r_1, r_3), (r_2, r_3)\}$. Množina potvrdených literálov $War(\mathcal{P}) = \{a, d, e, \neg c\}$ a množina všetkých stabilných modelov je $SM(\boldsymbol{P}) = \{M = \{e, d, a, c\}, War(\mathcal{P})\}$. Vidíme, že $\neg c \in War(\mathcal{P})$ a $c \in M$. Ukážeme, že $War(\mathcal{P})$ je preferovaný voči M. Podľa definície 20 $maxP_{War(\mathcal{P})}(\neg c) = 3$ a na druhej strane $maxP_M(c) = 1$, a teda $3 > max(1, 1)$.

Tvrdenie 3. *Nech* $\mathcal{P} = (\Pi, \Delta)$ *je vyvrátiteľný program,* $\boldsymbol{P} = (P_1, \dots, P_m)$ *jeho* π*-transformácia na postupnosť aktualizácií a W množina potvrdených literálov programu* $\mathcal{P}$*. Ak* $Args_{\mathcal{P}}$ *je množina len singulárnych argumentov,* $\prec$ *lineárne usporiadanie na* Δ*, v* $\mathcal{P}$ *nie je nepriamy konflikt, potom W je preferovaný model* $((\bigcup \boldsymbol{P} \setminus Reject(\boldsymbol{P}, W))^W)$*.*

Dôkaz. Zhrňme najprv ideu dôkazu: Za daných predpokladov, potvrdený literál nie je atakovaný žiadnym pravidlom a to sa prenáša aj do postupnosti aktualizácií. Ukážeme, že každý potvrdený literál patrí do preferovaného stabilného modelu a naopak, každý objektívny literál z preferovaného stabilného modelu je potvrdený literál.

[3] Uvedomme si, že $B(maxR_S(L))^+ \setminus F$ je maximálne jednoprvká množina, nakoľko uvažujeme len singulárne argumenty.

- „$\subseteq$" Ak $h \in War(\mathcal{P})$, potom $\exists \mathcal{A} = (A, h) \in Args_{\mathcal{P}}$, ktorého koreň vyhodnoteného dialektického stromu $\mathcal{T}^*_{\mathcal{A}}$ je označený ako „N". Bez ujmy na všeobecnosti nech $\mathcal{A} = maxArg(h)$. Na to, aby sme ukázali, že $h \in least^4((\bigcup \boldsymbol{P} \setminus Reject(\boldsymbol{P}, W))^W)$ musí $\exists r \in (\bigcup \boldsymbol{P} \setminus Reject(\boldsymbol{P}, W))^W$, $H(r) = h \ \wedge \ W \vDash B(r)$. Keďže pre h $\exists \mathcal{A} = (A, h) \in Args_{\mathcal{P}}$, potom existuje pravidlo $q \in \mathcal{P}, H(q) = h$. Vieme, že π-transformácia zabezpečuje, že $(\forall q \in \Delta)(\exists i \in \{1, \ldots, m-1\}), \pi(q) \in P_i$ a $P_m = \Pi$. Teda pre nami hľadané pravidlo r určite platí $r \in \bigcup \boldsymbol{P}, H(r) = h$. Ďalej si uvedomme, že $r \notin Reject(\boldsymbol{P}, W)$: keďže $\mathcal{A} = maxArg(h)$, podľa lemy 2 vieme, že $\nexists \mathcal{B} \in Args_{\mathcal{P}}, (\mathcal{B}, \mathcal{A}) \in \mathcal{R}$ a teda pre $P_i = \{\pi(A)\}$ $\nexists s \in P_j, j \in \{i+1, \ldots, m\}, H(r) = \neg H(s)$. Množina W zredukuje program $(\bigcup \boldsymbol{P} \setminus Reject(\boldsymbol{P}, W))$ o tie pravidlá $S = \{s \in P_i | i \in \{1, \ldots, m-1\}, B(s)^- \cap W \neq \emptyset\}$. Vieme, že $\forall q \in \Delta, B(\pi(q))^- = \neg H(q)$ a $\forall q \in \Pi, B(\pi(q))^- = \emptyset$. Na základe prechdádzajúcich pozorovaní potom dostaneme $r \notin S$. Skutočnosť, že $W \vDash B(r)$ už vyplýva z predpokladu $\mathcal{A} = (A, h) \in Args_{\mathcal{P}}$. Z predchádzajúceho a na základe def. preferovaného modelu potom dostaneme, že W je preferovaný stabilný model.
- „$\supseteq$" Nech $M = least((\bigcup \boldsymbol{P} \setminus Reject(\boldsymbol{P}, W))^W)$ je preferovaný stabilný model $\boldsymbol{P}$. Aby sme dostali $M \subseteq War(\mathcal{P})$, musí platiť $\forall h \in M$ $\exists maxArg(h) = \mathcal{A} = (A, h) \in Args_{\mathcal{P}}$, že koreň ohodnoteného dialektického stromu $\mathcal{T}^*_{\mathcal{A}}$ je označený ako „N". Z faktu, že $(\forall q \in \Delta)(\exists i \in \{1, \ldots, m-1\}), \pi(q) \in P_i$ a $P_m = \Pi$ a z úvodného predpokladu o množine M ďalej dostávame, že $(\forall h \in M)(\exists maxArg(h) = \mathcal{A} = (A, h) \in Args_{\mathcal{P}}) \ \wedge \ \nexists \mathcal{B} \in Args_{\mathcal{P}}, (\mathcal{B}, \mathcal{A}) \in \mathcal{R}$ a nakoniec podľa lemy 2 $M \subseteq War(\mathcal{P})$.

Skutočnosť, že W je stabilný model $\boldsymbol{P}$ teraz už vyplýva z definície 1.□

Uvedomme si nutnosť podmienky „*neprítomnosť nepriameho konfliktu v* $\mathcal{P}$" v predpokladoch tvrdenia 3. Nasledujúce príklady demonštrujú situáciu v prípade, že by sme od tejto podmienky upustili.

Príklad 4. Nech $\mathcal{P} = (\Pi, \Delta)$ je vyvrátiteľný program

$$\Pi = \left\{ \begin{array}{l} a. \\ \neg b \leftarrow c. \end{array} \right\} \qquad \Delta = \left\{ \begin{array}{l} r_1 : c \prec a. \\ r_2 : b \prec a. \end{array} \right\}$$

$\prec = \{(r_1, r_2)\}$ usporiadanie na vyvrátiteľných pravidlách. Ďalej máme argumenty $\mathcal{A} = (\{r_1\}, c)$, $\mathcal{B} = (\{r_2\}, b)$. $\mathcal{P}$ je π-transformovaný do postupnosti aktualizácií $\boldsymbol{P} = (P_1, P_2, P_3)$, kde:

$P_3 = \Pi = \{a. \ \neg b \leftarrow c.\}$
$P_2 = \pi(\{r_2\}) = \{b \leftarrow a, not\ \neg b.\}$
$P_1 = \pi(\{r_1\}) = \{c \leftarrow a, not\ \neg c.\}$

Vidíme, že $\mathcal{R} = \{(\mathcal{B}, \mathcal{A})\}$ a teda $War(\mathcal{P}) = \{a, b\}$, ale $SM(\boldsymbol{P}) = \{\{a, \neg b, c\}\}$ a teda $War(\mathcal{P}) \notin SM(\boldsymbol{P})$.Tvrdenie 3 v tomto prípade teda neplatí.

[4] Hoci bol operátor *least* pôvodne navrhnutý len pre definitné programy, kôli skráteniu rozsahu textu sme sa ho rozhodli použiť ako označenie minimálneho modelu aj pre nie-definitné programy.

Príklad 5. Uvažujme teraz vyvrátiteľný program $\mathcal{P}$ uvedený v príklade 2. Ďalej $\prec = \{(r_1, r_2)\}$ je usporiadanie na vyvrátiteľných pravidlách, $\mathcal{A} = (\{r_1\}, d)$, $\mathcal{B} = (\{r_2\}, c)$, $\mathcal{C} = (\{r_1\}, \neg f)$, $\mathcal{D} = (\{r_2\}, f)$ sú argumenty. $\mathcal{P}$ je π-transformovaný do postupnosti aktualizácií $\boldsymbol{P} = (P_1, P_2, P_3)$, kde:

$$P_3 = \Pi = \{a.\ b.\ f \leftarrow c.\ \neg f \leftarrow d.\}$$
$$P_2 = \pi(\{r_2\}) = \{c \leftarrow a, not\ \neg c.\}$$
$$P_1 = \pi(\{r_1\}) = \{d \leftarrow b, not\ \neg d.\}$$

Vidíme, že $\mathcal{R} = \{(\mathcal{B}, \mathcal{A}), (\mathcal{D}, \mathcal{A}), (\mathcal{D}, \mathcal{C}), (\mathcal{B}, \mathcal{C})\}$ a teda $War(\mathcal{P}) = \{a, b, c, f\}$, ale $SM(\boldsymbol{P}) = \emptyset$. Okrem skutočnosti, že tvrdenie 3 v tejto situácií neplatí, nám príklad demonštroval aj skutočnosť, že sémantika postupnosti aktualizácií, tak ako je zhrnutá v sekcii podľa [3], nie je schopná vysporiadať sa s nepriamymi konfliktami.

Uvažujme teraz iný prístup na transformáciu vyvrátiteľných pravidiel.

Definícia 21. *(ϕ-transformácia vyvrátiteľných pravidiel) Nech $\mathcal{P} = (\Pi, \Delta)$ je vyvrátiteľný program. Funkcia ϕ je ξ-transformácia, ktorá transformuje každé pravidlo z Δ v tvare $A \prec A_1, \ldots, A_n$ na pravidlo rozšíreného logického programu: $A \leftarrow A_1, \ldots, A_n$.*

Lemma 4. *Nech $\mathcal{P} = (\Pi, \Delta)$ je vyvrátiteľný program, $\boldsymbol{P} = (P_1, \ldots, P_m)$ jeho ϕ-transformácia na postupnosť aktualizácií a W množina potvrdených literálov programu $\mathcal{P}$. Ak $Args_{\mathcal{P}}$ je množina len singulárnych argumentov, $\prec$ lineárne usporiadanie na Δ, v $\mathcal{P}$ nie je nepriamy konflikt, potom $W = least((\bigcup \boldsymbol{P} \setminus Reject(\boldsymbol{P}, W))^W)$ je stabilný model.*

Dôkaz. Analogicky ako dôkaz tvrdenia 3. □

Tvrdenie 4. *Nech $\mathcal{P} = (\Pi, \Delta)$ je vyvrátiteľný program, $\boldsymbol{P} = (P_1, \ldots, P_m)$ jeho ϕ-transformácia na postupnosť aktualizácií a W množina potvrdených literálov programu $\mathcal{P}$. Ak $Args_{\mathcal{P}}$ je množina len singulárnych argumentov, $\prec$ lineárne usporiadanie na Δ, v $\mathcal{P}$ nie je nepriamy konflikt, potom $W = least((\bigcup \boldsymbol{P} \setminus Reject(\boldsymbol{P}, W))^W)$ je jediný stabilný model.*

Dôkaz. Na základe lemmy 4 je W stabilný model. Sporom nech existuje stabilný model S, $S \neq W$. To znamená, že $\exists L \in Obj$, $L \in S \wedge \neg L \in W$. Môžu nastať len 2 prípady, ktoré vedú k sporu[5]:

- $maxP_S(L) > maxP_W(\neg L)$. Na základe lemy 3 platí $\{h | \Pi \vdash h\} \subseteq W$ a podľa pozorovania 1 potom $W \models B(maxR_S(L))$ a teda aj $\forall s \in \boldsymbol{P}$, $H(s) = \neg L$ implikuje $s \in Rej(\boldsymbol{P}, W)$ (na základe def. 1). To je však spor s tým, že $\neg L \in W$.
- $maxP_S(L) < maxP_W(\neg L)$. Rovnakým spôsobom ako v predchádzajúcom bode vieme ukázať, že $\forall s \in \boldsymbol{P}$, $H(s) = L$ implikuje $s \in Rej(\boldsymbol{P}, S)$, čo je spor s tým, že $L \in S$.

[5] $maxP_S(L) = maxP_W(\neg L)$ nastať nemôže pretože v tvrdení neuvažujeme konflikt na rovnakej úrovni a predpokladáme linearitu $\prec$.

5 Záver

Študovali sme dva konkrétne formalizmy na reprezentáciu znalostí. Navhrhli sme dva typy transformácii vyvrátiteľného na dynamický program. Zaujímalo nás, v akom vzťahu je množina potvrdených literálov voči množine stabilných modelov transformovaného vyvrátiteľného programu. Zistili sme, že stabilno-modelová sémantika dynamického programu nie je schopná riešiť nepriamy konflikt, ktorý sme v príspevku uviedli. Tvrdenia 3 a 4 považujeme za hlavné výsledky práce.

Literatúra

1. G. Brewka *On the relationship between defeasible logic and well-founded semantics,* Proc. LPNMR 2001, Springer LNAI 2173, pages 121-132, 2001, ISBN:3-540-42593-4.
2. P.M. Dung *On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games,* Artificial Intelligence, Vol 77, No 2, Elsevier Publishing B.V, 1995.
3. T. Eiter, G. Sabbatini, M. Fink, H. Tompits *On updates of logic programs: semantics and properties,* INFSYS Research Report 1843- 00-08, 2001.
4. A. J. García, G. R. Simari *Defeasible Logic Programming An Argumentative Approach* Journal of Theory and Practice of Logic Programming. Vol 4(1-2) pages 95-138, 2004, ISSN:1471-0684.
5. M. Thimm, G. Kern-Isberner *On the relationship of defeasible argumentation and answer set programming,* In Proceedings of the Second International Conference on Computational Models of Argument (COMMA'08), pages 393-404, 2008.

Annotation:

Title: Conflicts solving in defeasible and dynamic logic programs

Nowadays, argumentation formalisms are in the spotlighit in knowledge representation area. We studied defeasible logic program and update sequence formalisms and focused on way how they resolve conflicts. Two trasformations for defeasible program are proposed. An indirect conflict is introduced and some involving problems are discussed. We show, under what circumstances, set of warranted literals is stable model of update sequence.