Semantics for Conditional Literals via the SM Operat

Zach Hansen Yuliya Lierler

University of Nebraska Omaha

September 2022

Table of Contents

Motivation

Program Veri cation

Running Example

Table of Contents

Program Syntax Continued

[redacted]

For a signature = (O; F; P) of a rst-order language:

- O is the set of object constants;
- F is the set of function symbols (non-zero arity);
- P is the set of predicate constants;
- G is the set of all ground terms constructed from O and F of .

For a program :

- = (O ; F ; P), where O , F , and P contain all the object constants, function symbols, and predicate constants occurring in ;
- G denotes $G_{(O;F;P)}$.

Table of Contents

Global vs. Local Variables

Global vs. Local Variables

- A variable is *global* in a conditional literal $H : L$ if it occurs in H but not in L;
- All other variables occurring in a conditional literal are *local*.
- All variables are global in basic literals;
- A variable is global in a rule if it is global in at least one rule literal.

Example

In rule

$$\text{:- not } asg(V, I) : color(I); vtx(V).$$

with conditional literal

$$\text{not } asg(V, I) : color(I);$$

V is a global variable, whereas I is local.

Syntactic Transformation

z

For a rule R with global variables z:

- for a conditional literal H : L occurring in the body of R with local variables x, $_z(H : L)$ is 8x ($_z(L)$! $_z(H)$) ;
- for a conditional literal H : L occurring in the head of R with local variables x, $_z(H : L)$ is 9x ($_z(L)$! $_z(H)$) ^ :: $_z(L)$:

(R) is the formula 8z($_z(B_1)$ ^ ^ $_z(B_n$! $_z($

Graph Coloring Example

Conditional Literal Translation

Transformation applied to the conditional literal

not asg(V ; I) : color(I)

produces formula

8i (color(i) ! : asg(v; i))

( )

8vi (vtx(v) ^ color(i)) ! asg(v; i) _ : asg(v; i)

8v (8i (color(i) ! : asg(v; i)) ^ vtx(v)) ! ?

8vij (asg(v; i) ^ asg(v; j) ^ i 6= j ^ vtx(v) ^ color(i) ^ color(j)) ! ?

8viw (asg(v; i) ^ asg(w; i) ^ vtx(v; w) ^ color(i) ^ edge(v; w)) ! ?

Semantics of Conditional Logic Programs

Semantics via the SM operator

An interpretation is a p-stable model of a rst-order sentence F when it is a model of $SM_p[F]$.

For a conditional logic program and a Herbrand interpretation over the signature (O ; F ; P), I is an answer set of when I is a P -stable model of ( ).

Conditional Programs via Infinitary Propositional Logic

Traditional Characterization of Conditional Literals:

(i) Syntactic reduction to IPL formulas;[1]

(ii) Semantics defined by IPL stable model semantics.[2]

[redacted]

If x is the list of variables occurring in a conditional literal $l : L$:

Body:

Rules to Closed Rules

[redacted]

$\mathrm{inst}_G(R) = \ f\, R^z_u \ j \ u \ 2 \ G^z g^\wedge$

[redacted]

$$(R) = \ f \quad (r) \ j \ r \ 2 \ \mathrm{inst}_G \ (R) g^\wedge :$$

Similarly,

$$(\ ) = \quad f \quad (R) \text{ w.r.t. } G \quad j \ R \ 2 \quad g^\wedge :$$

Connecting Semantics for Conditional Literals

[redacted]

- We have presented our translation, which produces nonground rst-order formulas from logic programs.
- We have reviewed translation, which produces in nitary propositional formulas from logic programs.
- We use a previously established result to show that the P-stable models of ( ) coincide with the stable models of ( ).

Grounding

[redacted]

- gr(?) = ? ;
- gr(A) = A for a ground atomA;
- gr(t_1 = t_2) is > if t_1 is identical tot_2, and ? otherwise, for ground terms t_1; t_2;
- If F = G _ : : : _ H, then gr(F) = gr(G) _ _ gr(H);
- If F = G ^ : : : ^ H, then gr(F) = gr(G) ^ ^ gr(H);
- If F = G ! H, then gr(F) = gr(G) ! gr(H);
- If F = 9xG(x), then gr(F) = f gr(G(u)) j u 2 G^x g_ ;
- If F = 8xG(x), then gr(F) = f gr(G(u)) j u 2 G^x g^ .

Graph Coloring Example

[redacted]

Take rule R to be :- not asg(V ; I) : color(I); vtx(V): The grounding of (R) replaces global variables:

gr((R)) = f gr (8i (color(i) ! : asg(1; i)) ^ vtx(1)) ! ? ;

gr (8i (color(i) ! : asg(g; i)) ^ vtx(g)) ! ? g^

Take closed rule r to be :- not asg(1; I) : color(I); vtx(1): Then,

(r) = (color(1) ! : asg(1; 1)) ^ (color(g) ! : asg(1; g)) ^ vtx(1) ! ?

[redacted]

(8i (color(i) ! : asg(1; i)) ^ vtx(1)) ! ?

is equivalent to

(color(1) ! : asg(1; 1)) ^ (color(g) ! : asg(1; g)) ^ vtx(1) ! ?

Conclusion

[redacted]