Localizing Uniquely Determined Programs

Pascal Hitzler and Anthony Seda

MFCSIT2000, Cork, July 2000

Contents

Dept. of Mathematics, University College, Cork, Ireland
{phitzler,aks}@ucc.ie http://maths.ucc.ie/∼{pascal,aks}/

Logic Programs

A logic program P is a finite set of clauses

$$\forall(A \leftarrow L_1 \wedge \cdots \wedge L_n)$$

from first order logic usually written as

$$A \leftarrow L_1, \ldots, L_n \quad \text{or as} \quad A \leftarrow \texttt{body},$$

where A an atom, L_i a literal, $n \geq 0$.
B_P: Herbrand base.
$I_P = 2^{B_P}$: set of all Herbrand interpretations.
ground(P): set of all ground clauses of P.

Define (nonmonotonic) operator $T_P : I_P \to I_P$ by
$T_P(I)$ is set of all $A \in B_P$
for which there is a clause $A \leftarrow L_1, \ldots, L_n$
in ground(P) s.t. $I \models L_1 \wedge \cdots \wedge L_n$.

I is a model iff $T_P(I) \subseteq I$.

I is a *supported model* iff $T_P(I) = I$.
(introduced in Apt, Blair & Walker 1988)

P is *uniquely determined*
if it has unique supported model.

Why Study Uniquely Det. Progs?

Literature: Many classes of programs known,
e.g. having certain termination properties
or semantical properties.

Termination

Prolog systems compute by constructing
proof trees from programs.
Tree dependent on a *selection function* s.

Each s gives rise to a different class
of programs which terminate under s,
e.g. acyclic, acceptable, fair-bounded progs.
These programs are often uniquely determined.

On the other hand,
uniquely determined progs can implement
all partial recursive functions.

So we have the full range from
strong termination properties
to computational adequacy.

Why Study Uniquely Det. Progs?

Denotational Semantics

Logic Programming is
a simple model of reasoning.

Logic programs are axiom systems
in first order logic.

Semantics (“meaning”) of a program P is given
by designated (“intended”) models of P.

(E.g. intended models are the supported ones.)

Literature: Many semantics known,
differing on the interpretation of “intended”.

- How can the meaning of P be
  - $*$ characterized?
  - $*$ obtained?
- To what extent are
  uniquely determined programs
  unambiguous?

Classes of Uniquely Determined Programs

level mapping $l : B_P \to \gamma$ (γ ordinal)

P *locally hierarchical* (*lh*) if $l(A) > l(L_i)$ for each $A \leftarrow L_1, \ldots, L_n$ in ground(P) and each i.

If P is lh then P has unique supported model.
We want to weaken the defining condition.

P logic program, I model for P, l level mapping.

P is Φ^*-accessible ($P \in [\Phi^*]$) if
for each $A \leftarrow L_1, \ldots, L_n$ in ground(P) either
$I \models L_1 \wedge \cdots \wedge L_n$ and $L(A) > l(L_i)$ for all i or
exists i s.t. $I \not\models L_i$ and $l(A) > l(L_i)$.

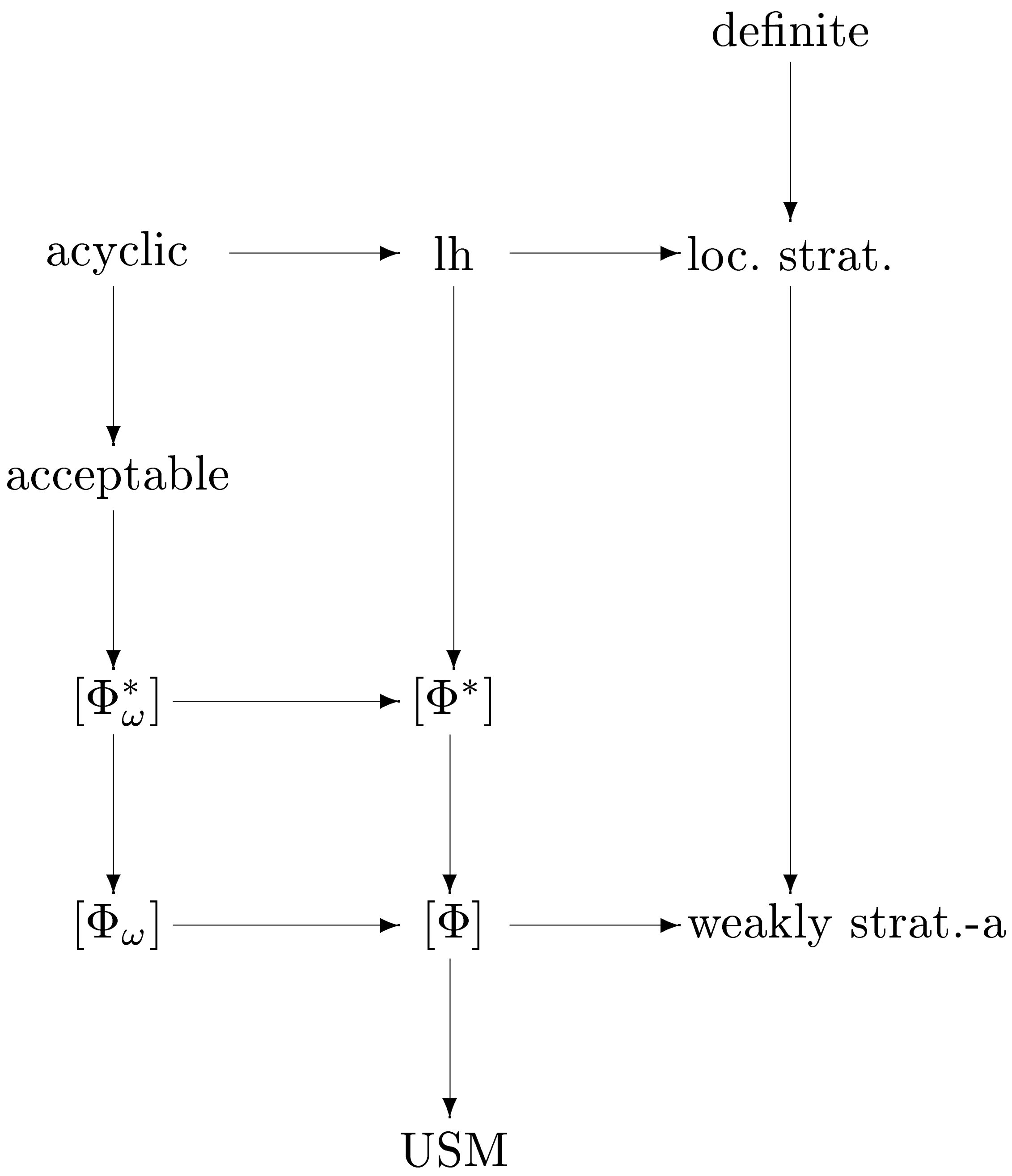
Cf. Apt & Pedreschi 1993: Acceptable Programs.

P is Φ-accessible ($P \in [\Phi]$) if
each $A \in B_P$ satisfies either (i) or (ii):

(i) Exists $A \leftarrow L_1, \ldots, L_n$ in ground(P) s.t.
$I \models L_1 \wedge \cdots \wedge L_n$ and $l(A) > l(L_i)$ for all i.

(ii) For each $A \leftarrow L_1, \ldots, L_n$ in ground(P)
exists i with $I \not\models L_i$, $I \not\models A$, $l(A) > l(L_i)$.

Classes of Uniquely Determined Programs



Generalized Metric Approaches

Cast I_P into generalized metric space s.t. fixed-point theorems can be applied to T_P.

P	I_P	Theorem
acyclic	metric	Banach
lh	gum	P-C & R
$[\Phi_\omega]$	d-metric	Matthews
$[\Phi]$	d-gum	H & S

gum: generalized ultrametric,
distance function maps into poset

P-C & R: Priess-Crampe & Ribenboim 199x

d-metric: dislocated metric,
$d(x, x) \neq 0$ allowed

Matthews: 1985
(*partial metrics* 199x)

d-gum: dislocated generalized ultrametric,
merges d-metric and gum

H & S: Hitzler & Seda 2000

Fitting-style Semantics

Let $\mathcal{L}$ be a 3-valued logic (f,u,t), P a program.
$I_{P,3}$ denotes set of all $I = (I^+, I^-)$, with
$I^+, I^- \subseteq B_P$, $I^+ \cap I^- = \emptyset$.

$\mathcal{F} : I_{P,3} \to I_{P,3}$ defined by $\mathcal{F}(I) = (T, F)$, where
* T is set of all A s.t. exists $A \leftarrow$ `body` with `body` true (t) in I,
* F is set of all A s.t., for all $A \leftarrow$ `body`, `body` is false (f) in I.

Corresponding $\mathcal{F}^*$ def. by $\mathcal{F}(I) = (T, F)$, where
* T is set of all A s.t. for all $A \leftarrow$ `body` `body` is true (t) in I,
* F defined as for $\mathcal{F}$.

If $\mathcal{L}$ is Kleene's strong 3-valued logic ($\mathcal{F} = \Phi$, introduced in Fitting 1985) then
- $P \in [\Phi]$ iff $\text{lfp}(\Phi) = (K, B_P \setminus K)$. K unique supported model of P.

By changing $\mathcal{L}$, acyclic, lh, acceptable and other programs can be characterized analogously.

For acceptable programs, $\mathcal{L}$ is not commutative.

Relations to Other Semantics

Let P be Φ-accessible,
M its unique supported model.

- M minimal (*not* least) 2-valued model.
- $(M, B_P \setminus M)$ unique Kripke-Kleene model. (Fitting 1985, 1996)
- M unique stable model. (Gelfond & Lifschitz 1988, 1991) (Answer Set Programming, 1997)
- M total well-founded model. (Van Gelder, Ross & Schlipf 1991)
- M weakly-perfect-a model. (Przymusinska & Przymusinski 1990)
- M is a limit of transfinite iterations of T_P.

∗ Neither of these results can be generalized to USM.

Let P be lh, M its unique supported model.

- M unique perfect model. (Przymusinski 1989, 199x cf. stratified progs, Apt, Blair & Walker)

∗ Does not hold in general for acceptable P.

Stable and Supported Models

$I \in I_P$ is *well-supported* (Fages 1994) if
exists strict well-founded partial order $\prec$ on I
s.t. for any $A \in I$ exists
$A \leftarrow B_1, \ldots, B_n, \neg C_1, \ldots, \neg C_m$ in ground(P)
with $I \models B_1 \wedge \ldots \wedge B_n \wedge \neg C_1 \wedge \cdots \wedge \neg C_m$
and $B_i \prec A$ for each i.

Let P be a logic program.
M is well-supported model iff M is stable model.

Let P' be obtained from P by omitting
all negative literals in the clauses.

Let P be program s.t. P' is Φ^*-accessible.
Then M supported model iff M stable model.

$*$ Result does not generalize to $[\Phi]$.

Discussion and Further Work

Generalized Metric Approaches

- Relations to neural networks.
  (Hölldobler et al. 199x)
  (Hitzler & Seda 2000)
- Relations to topological dynamics.
  (Hitzler & Seda 1997)
- Continuous methods.
  (Blair et al. 1999)
- Power Default Reasoning.
  (Rounds & Zhang 199x)

Fitting-style Semantics

- Characterizations of larger classes of programs.
- Termination analysis of
  other multivalued logics.

Relations to Other Semantics

- Examine the “space” of all logic programs.

Stable and Supported Models

- Answer set programming.
  (DeReS, Smodels, DLV, CCALC etc. 1997)