PDDL Axioms Are Equivalent to Least Fixed Point Logic

Claudia Grundke Gabriele Röger

University of Basel

July 2, 2026

Axioms and LFP

PDDL axioms: logic rules to derive truth values of state variables

$$\mathit{path}(x,y) \leftarrow E(x,y) \lor \exists z (E(x,z) \land \mathit{path}(z,y))$$
$$\mathit{acyclic}() \leftarrow \forall x \, \neg \mathit{path}(x,x)$$

Axioms and LFP

PDDL axioms: logic rules to derive truth values of state variables

$$\mathit{path}(x, y) \leftarrow E(x, y) \vee \exists z (E(x, z) \wedge \mathit{path}(z, y))$$
$$\mathit{acyclic}() \leftarrow \forall x \, \neg \mathit{path}(x, x)$$

least fixed point logic: first-order logic extended with fixed point operator

$$\varphi(\mathit{path}, x, y) := E(x, y) \vee \exists z (E(x, z) \wedge \mathit{path}(z, y))$$
$$\psi_{\mathit{acyclic}} := \forall z \neg [\mathbf{lfp}_{\mathit{path}, x, y} \varphi(\mathit{path}, x, y)](z, z)$$

Expressible Queries

axiom query:

Is *acyclic()* true in state *s*?

$\equiv$

LFP query:

Does state *s* satisfy $\psi_{acyclic}$?

Expressible Queries

axiom query:

Is *acyclic()* true in state *s*?

≡

LFP query:

Does state *s* satisfy $\psi_{acyclic}$?

our main result:

Axioms and LFP can express exactly the same queries.

proof via compilations to LFP_0 and LFP-sim

Implications of Axioms ≡ LFP

Stratified axioms and PDDL standard axioms are **equivalent**.
⇒ compilation possible

Axioms in stratified Datalog are **less expressive**.
⇒ compilation impossible

Limitation of Methodology

We analysed **if** a language can express a query,

not how compactly it can express it.

Axioms ≡ LFP

⬇

Stratified Axioms ≡ PDDL Standard Axioms

Two Axiom Languages

stratified axioms
Thiébaux, Hoffmann, Nebel. In Defense of PDDL Axioms (AIJ, 2005)

allows any stratifiable set of axioms
an axiom may contain $\neg P$ only if P is defined in a lower stratum

PDDL standard
Edelkamp & Hoffmann. PDDL2.2: The Language for the Classical Part of the 4th International Planning Competition (2004, TR Uni Freiburg)

forbids negative occurrences of derived predicates in axioms bodies

Compiling Stratified Axioms Into PDDL Standard

compilation:

Eliminate negated derived predicates and merge strata into one.

key idea:

Make each stratum's fixed point computation explicit
by adding axioms that define the relation atom A is derived before atom B.

These axioms do not introduce new negated derived predicates!

idea adjusted from least fixed point logics:

Moschovakis (1974): transformation for infinite structures
Immerman (1986) and Gurevich (1986): adaptation to finite structures

Axioms $\equiv$ LFP

⬇

Axioms $>$ Stratified Datalog

Axioms Are More Expressive Than Stratified Datalog

stratified Datalog: rule bodies are existentially quantified **conjunctions of literals**

Kolaitis (1991)
Least fixed point logic can express queries that stratified Datalog cannot.

stratified Datalog cannot express:

$$\mathit{win}(x) \leftarrow \exists y(\mathit{move}(x, y) \land \forall z(\mathit{move}(y, z) \rightarrow \mathit{win}(z)))$$

Alice wins from game position x if she can move to a position y from which Bob cannot win.

Limitation of Fast Downward

Fast Downward's translator compiles stratified axioms into stratified Datalog form.

impossible on the lifted level (for some axioms)

for example:

- $\mathit{win}(x) \leftarrow \exists y(\mathit{move}(x, y) \wedge \forall z(\mathit{move}(y, z) \rightarrow \mathit{win}(z)))$
- spanning-tree by Ivankovic & Haslum (IJCAI 2015)

Limitation of Fast Downward

Fast Downward's translator compiles stratified axioms into stratified Datalog form.

impossible on the lifted level (for some axioms)

possible if we use the finite universe of a planning task

Summary

LFP ≡ stratified axioms ≡ PDDL standard axioms > stratified Datalog

⇒ We can compile stratified axioms into PDDL standard axioms but not into stratified Datalog (without task information).

link to the paper

Relationships to Fixed Point Logics

Libkin (2004), Corollaries 10.8 & 10.13:

$\text{LFP-sim} = \text{LFP} = \text{LFP}_0$

Trivial: $\text{AP}_0 \leq \text{AP}$

New: $\text{LFP}_0 \leq \text{AP}_0, \quad \text{AP} \leq \text{LFP-sim}$

Thus: $\text{AP} = \text{AP}_0$

Ebbinghaus and Flum (1995), Theorems 7.7.2 & 8.1.1:

$\text{Dat} = \text{BFP}, \quad \text{BFP} < \text{LFP}$

Thus: $\text{Dat} < \text{AP}, \quad \text{Dat} < \text{AP}_0$

AP = stratified axioms
AP_0 = PDDL standard axioms
Dat = stratified Datalog

LFP = least fixed point logic
LFP-sim = LFP with simultaneous fixed points
LFP_0 = first-order logic extended with single fixed point
BFP = bounded LFP

Libkin, L. (2004). Elements of Finite Model Theory. Springer Berlin, Heidelberg.
Ebbinghaus, H.-D. and Flum, J. (1995). Finite Model Theory. Springer-Verlag.