Corollaries on the Fixpoint Completion

Pascal Hitzler

International Center for Computational Logic, TU Dresden

Contents

The fixpoint completion of a logic program allows to transform Gelfond-Lifschitz operators (stable semantics) into simpler single-step operators (supported model semantics).

We study some corollaries from this observation.

Background

[DK89] Phan M. Dung and Kanchana Kanchanasut, A fixpoint approach to declarative semantics of logic programs. In: Ewing L. Lusk and Ross A. Overbeek, Logic Programming, Proceedings of the North American Conference 1989, NACLP'89, Cleveland, Ohio, MIT Press, 1989, pp. 603-625.

Program transformation $P \mapsto \text{fix}(P)$.
Complete unfolding through positive body literals.

[Wen02] Matthias Wendt, Unfolding the well-founded semantics, Journal of Electrical Engineering 53 (12/s), 2002, 56-59.

Shows $\text{GL}_P(I) = T_{\text{fix}(P)}(I)$ for all interpretations I.

The Fixpoint Completion

Quasi-interpretation Q: set of clauses of form $A \leftarrow \neg B_1, \ldots, \neg B_m$.

Program P: set of (ground) clauses of form

$$A \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m.$$

$T'_P(Q)$ set of $A \leftarrow \texttt{body}_1, \ldots, \texttt{body}_n, \neg B_1, \ldots, \neg B_m$
where $A \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in P
and $A_i \leftarrow \texttt{body}_i$ in Q for all i.

$T'_P \uparrow \omega = \text{lfp}(T'_P) = \text{fix}(P)$ quasi-interpretation.

Semantic Operators

$T_P(I)$ set of all A
with $A \leftarrow L_1, \ldots, L_n$ in P and $I \models L_1, \ldots, L_n$.

P/I set of all $A \leftarrow A_1, \ldots, A_n$
with $A \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ in P
and $I \not\models B_1, \ldots, I \not\models B_m$.

$\mathrm{GL}_P(I) = \mathrm{lfp}(T_{P/I})$.

For all interpretations I: $\mathrm{GL}_P(I) = T_{\mathrm{fix}(P)}(I)$. [Wen02]

In fact even: $\Psi_P(I) = \Phi_{\mathrm{fix}(P)}(I)$.

Iterative Behaviour

P locally hierarchical, i.e. exists level mapping l
with $l(A) > l(L_i)$ for all $A \leftarrow L_1, \ldots, L_n$ and all i.

Then T_P contraction (wrt. some generalized metric).
$T_P^\alpha(I)$ converges to unique supported model of P (for all I).

l maps to $\mathbb{N}$: T_P contraction with respect to metric.
l injective: T_P contraction on Cantor set (via isometry).

[Hitzler and Seda, Theoretical Computer Science 2003]

Iterative Behaviour

P locally stratified, i.e. exists level mapping l
with $l(A) \geq l(A_i)$ and $l(A) > l(B_j)$
for all $A \leftarrow A_1, \ldots, A_n, \neg B_1, \ldots, \neg B_m$ and all i.

Then trivially(!): $\mathrm{fix}(P)$ locally hierarchical.

$\mathrm{GL}_P \equiv T_{\mathrm{fix}(P)}$

GL_P contraction (wrt. some generalized metric).
$\mathrm{GL}_P^\alpha(I)$ converges to unique stable model of P (for all I).

l maps to $\mathbb{N}$: GL_P contraction with respect to metric.
l injective: GL_P contraction on Cantor set (via isometry).

Negation: Cantor topology

$\mathcal{G} = \{\mathcal{G}(A) : A \in B_P\} \cup \{\mathcal{G}(\neg A) : A \in B_P\}$ with $\mathcal{G}(L) = \{I \in I_P : I \models L\}$
Subbase of the *Cantor topology* Q on I_P.

B_P countable, then (I_P, Q) homeomorphic to the *Cantor set* $\mathcal{C}$ on $\mathbb{R}$.

Batarekh & Subrahmanian 1989 (*query topology*)
Seda 1995 (*atomic topology*)

$\lim T_P^n(I)$ is model of P, if existent. [Hitzler & Seda, SumTopo 1997]

Connectionist Systems

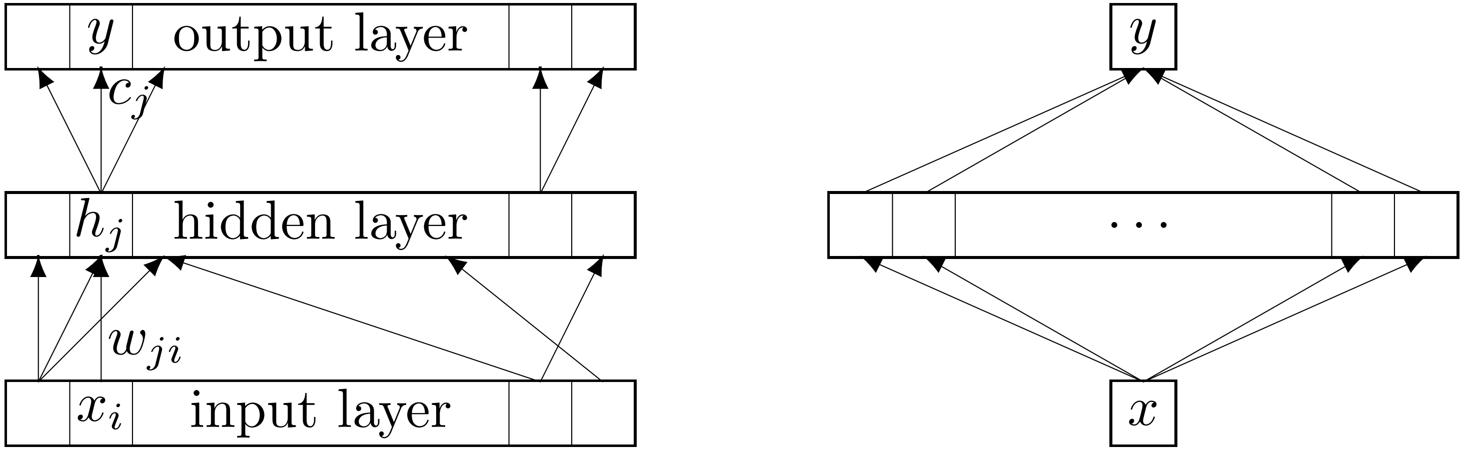
T_P continuous in Cantor topology,
then T_P uniformly approximable by artificial neural network.

P without local variables, then $\text{fix}(P)$ without local variables.
Then $T_{\text{fix}(P)}$ continuous [Seda 1995]. Hence GL_P continuous,
i.e. GL_P approximable by artificial neural network.

[Hitzler, Hölldobler and Seda, Journal of Applied Logic 2004]

Characterization of continuity of GL_P can be found in paper.

3-Layer Feedforward Networks



x_i inputs; y outputs; w_{ji}, c connection weights

I/O-function:

$$y = f(x_1, \ldots, x_r) = \sum_j c_j \phi \left(\sum_i w_{ji} x_i - \theta_j \right)$$

θ_j thresholds

σ threshold function (sigmoidal or gaussian)

Self-Similarity

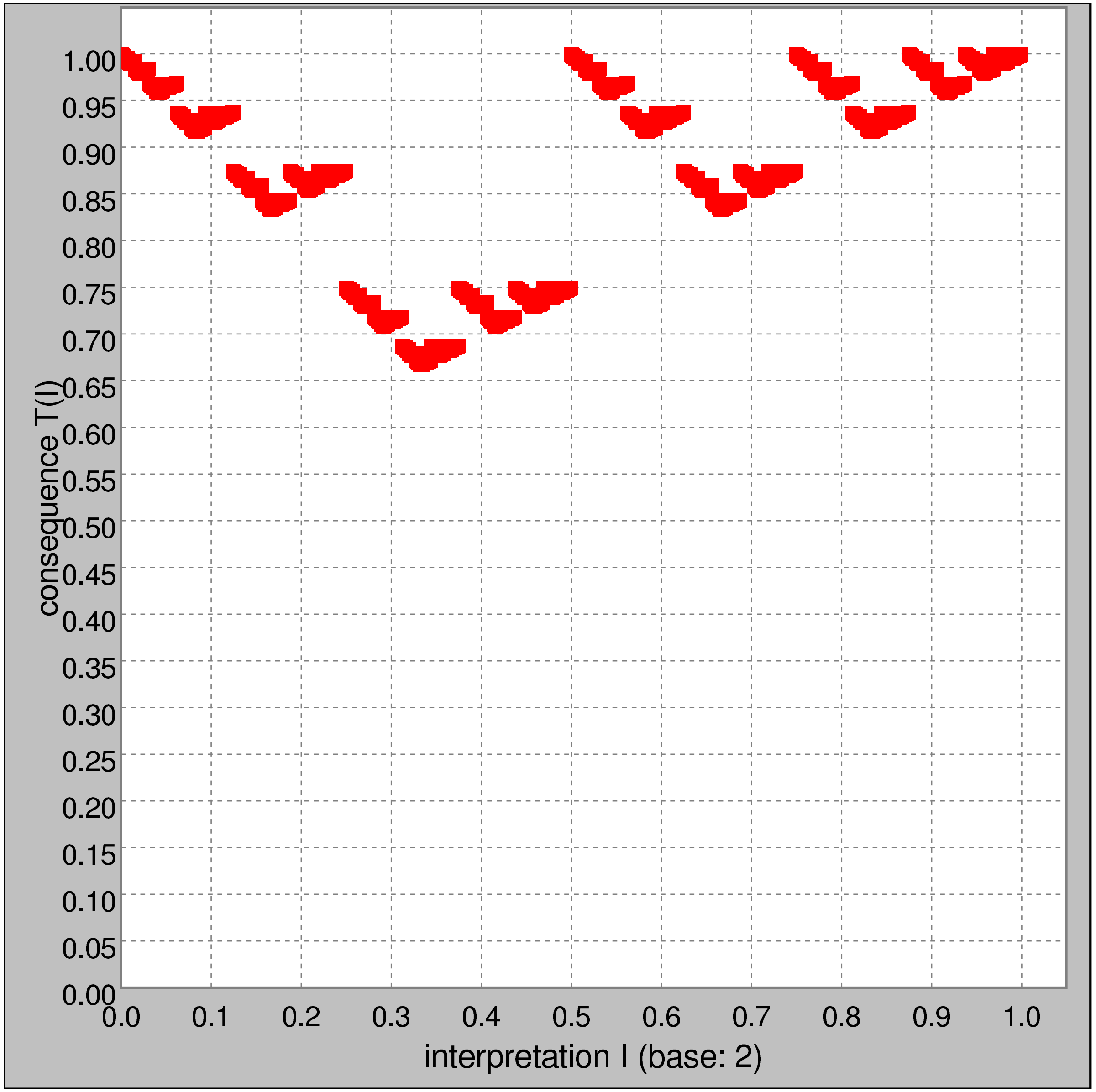
An obseration by Sebastian Bader.

Graph of T_P visualized via embedding into $[0,1] \times [0,1]$
using p-adic numbers.

$R : I_P \to \mathbb{R} : I \mapsto \sum_{A \in I} B^{-l(A)}$, where $l : B_P \to \mathbb{N}$ injective, $B > 2$.

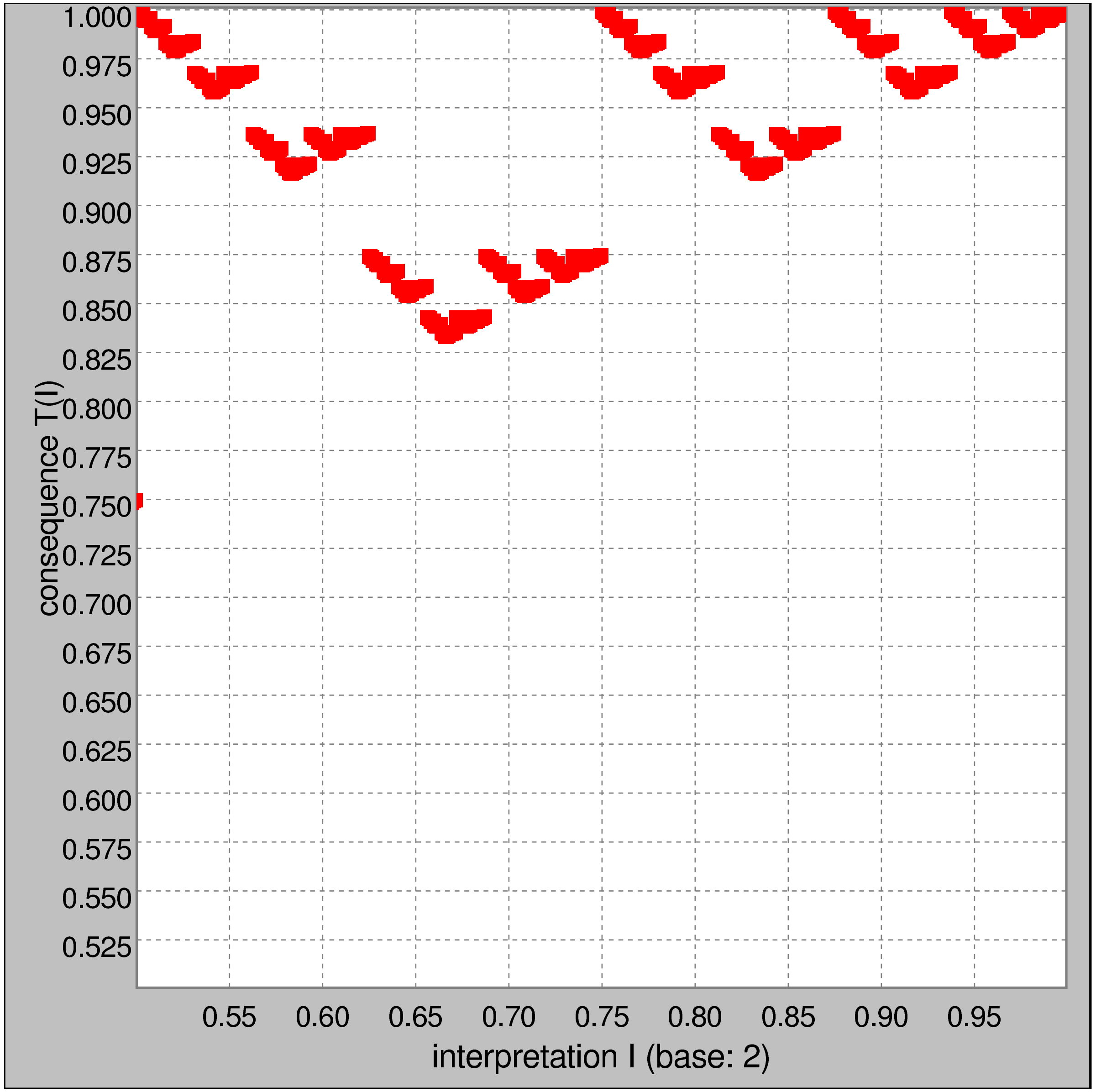
Graph shows self-similarity.

(The following pictures were provided by Sebastian Bader.)



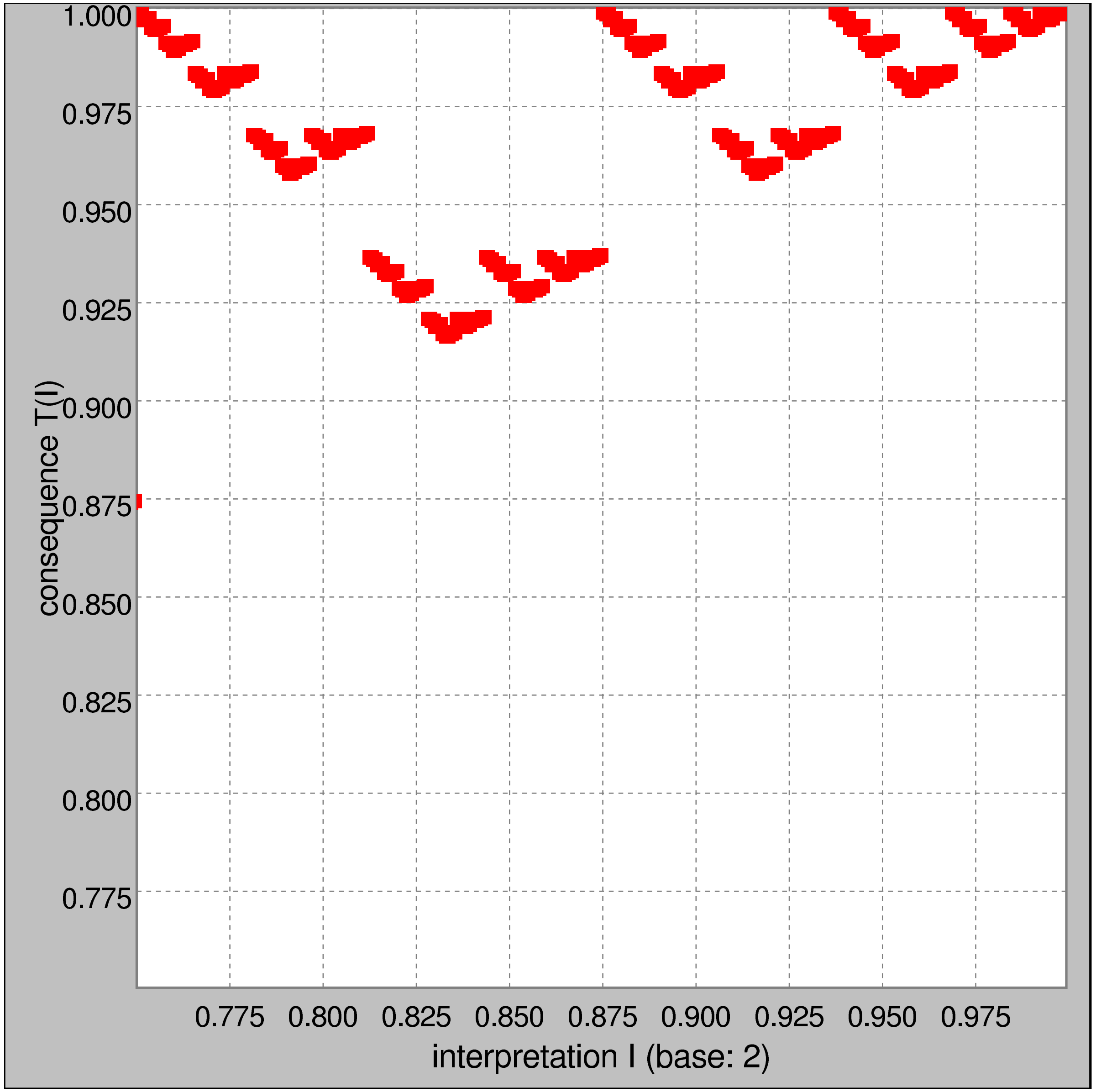

```
p(0).
p(s(X)) :- p(X).
p(X) :- not p(X).
```

```
p(s(s(s(s(s(s(s(s(s(s(0))))))))))) => 11
p(s(s(s(s(s(s(s(s(s(0)))))))))) => 10
p(s(s(s(s(s(s(s(s(0))))))))) => 9
p(s(s(s(s(s(s(s(0)))))))) => 8
p(s(s(s(s(s(s(0))))))) => 7
p(s(s(s(s(s(0)))))) => 6
p(s(s(s(s(0))))) => 5
p(s(s(s(0)))) => 4
p(s(s(0))) => 3
p(s(0)) => 2
p(0) => 1
```



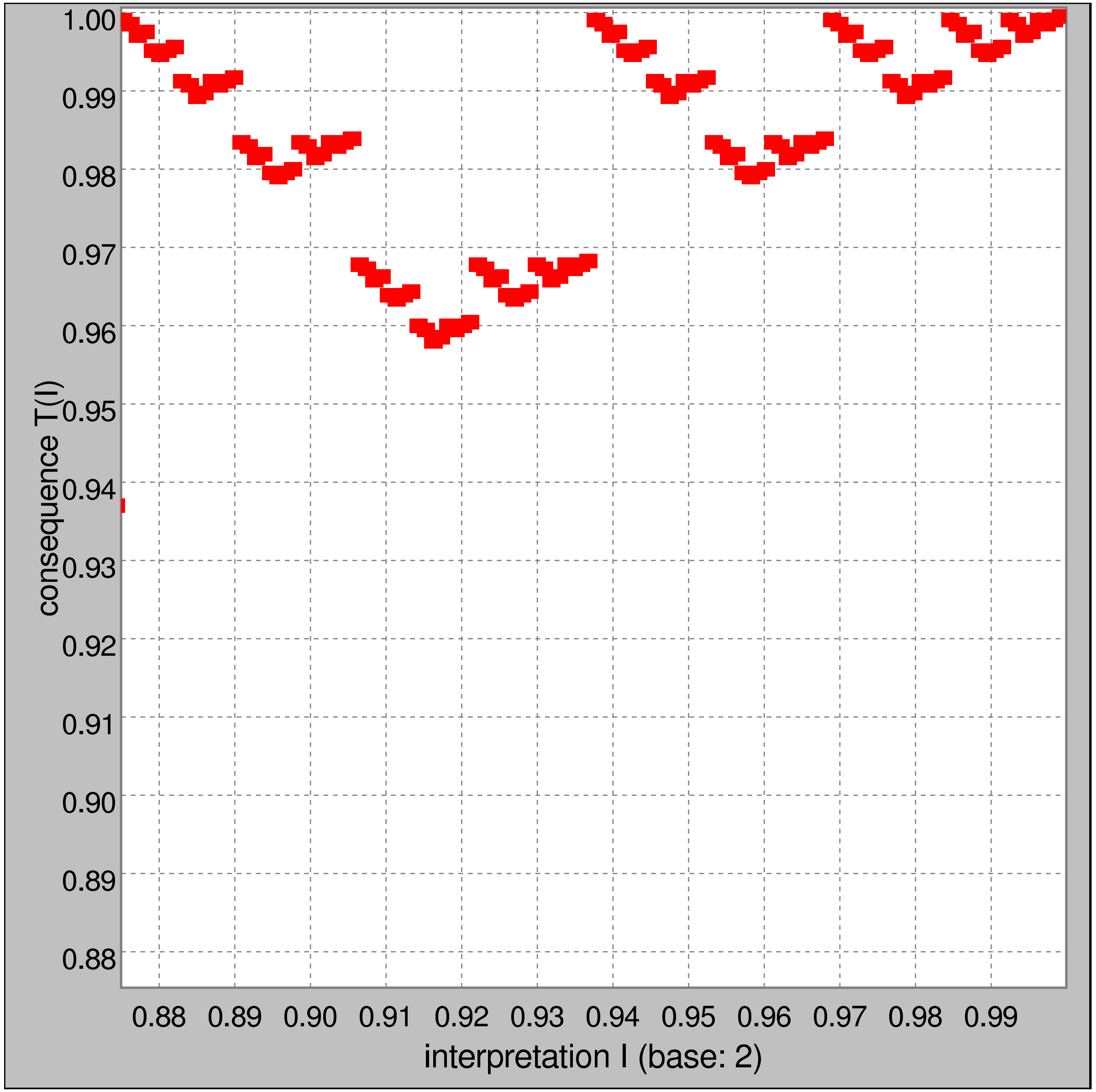

```
p(0).
p(s(X)) :- p(X).
p(X) :- not p(X).
```

```
p(s(s(s(s(s(s(s(s(s(s(0)))))))))) => 11
p(s(s(s(s(s(s(s(s(s(0))))))))) => 10
p(s(s(s(s(s(s(s(s(0)))))))) => 9
p(s(s(s(s(s(s(s(0))))))) => 8
p(s(s(s(s(s(s(0)))))) => 7
p(s(s(s(s(s(0))))) => 6
p(s(s(s(s(0))))) => 5
p(s(s(s(0)))) => 4
p(s(s(0))) => 3
p(s(0)) => 2
p(0) => 1
```



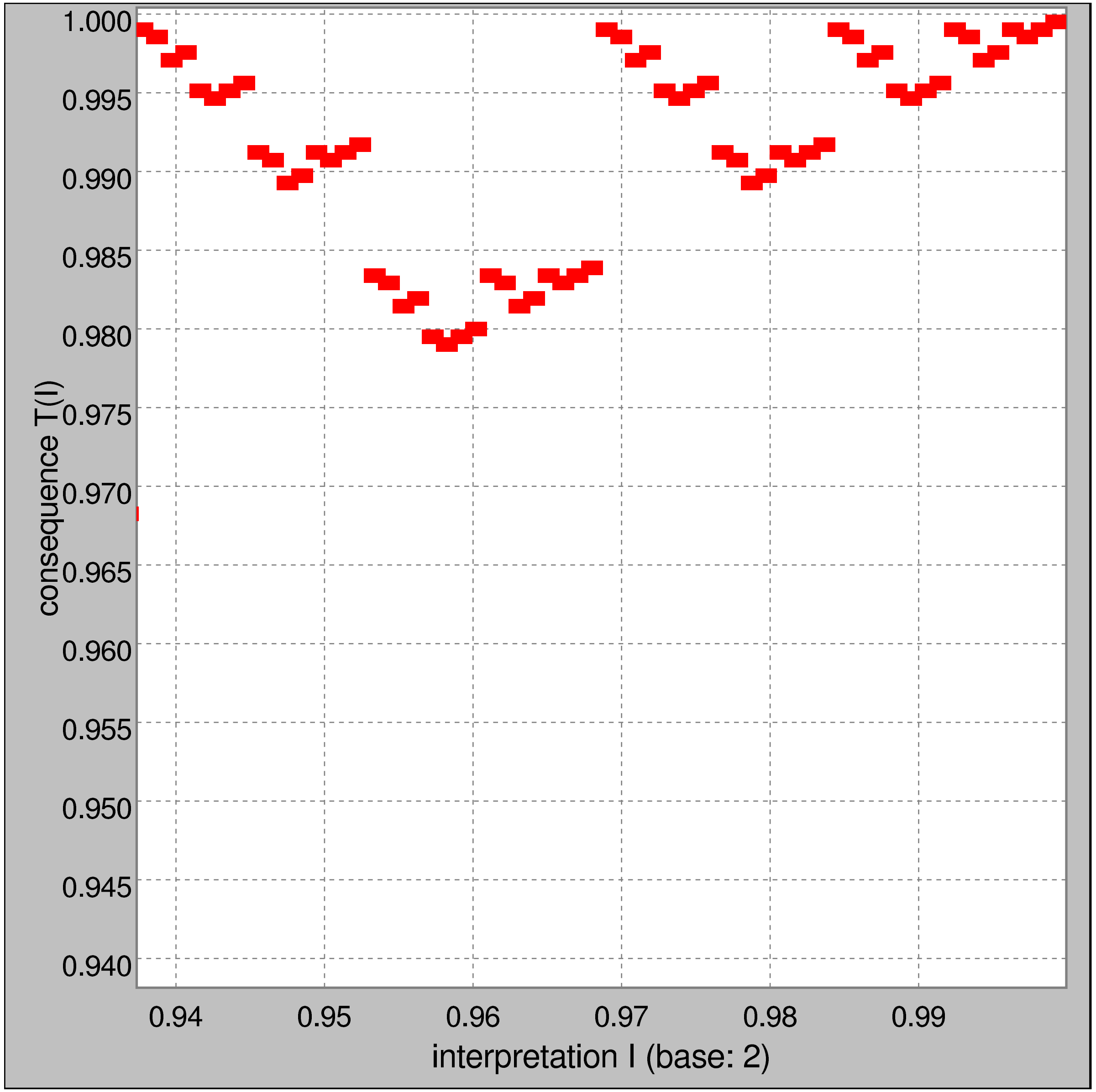

```
p(0).
p(s(X)) :- p(X).
p(X) :- not p(X).
```

```
p(s(s(s(s(s(s(s(s(s(s(0))))))))))) => 11
p(s(s(s(s(s(s(s(s(s(0)))))))))) => 10
p(s(s(s(s(s(s(s(s(0))))))))) => 9
p(s(s(s(s(s(s(s(0)))))))) => 8
p(s(s(s(s(s(s(0))))))) => 7
p(s(s(s(s(s(0)))))) => 6
p(s(s(s(s(0))))) => 5
p(s(s(s(0)))) => 4
p(s(s(0))) => 3
p(s(0)) => 2
p(0) => 1
```




```
p(0).
p(s(X)) :- p(X).
p(X) :- not p(X).
```

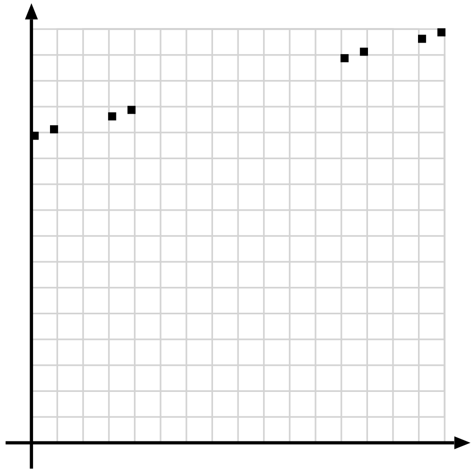
```
p(s(s(s(s(s(s(s(s(s(s(0))))))))))) => 11
p(s(s(s(s(s(s(s(s(s(0)))))))))) => 10
p(s(s(s(s(s(s(s(s(0))))))))) => 9
p(s(s(s(s(s(s(s(0)))))))) => 8
p(s(s(s(s(s(s(0))))))) => 7
p(s(s(s(s(s(0)))))) => 6
p(s(s(s(s(0))))) => 5
p(s(s(s(0)))) => 4
p(s(s(0))) => 3
p(s(0)) => 2
p(0) => 1
```



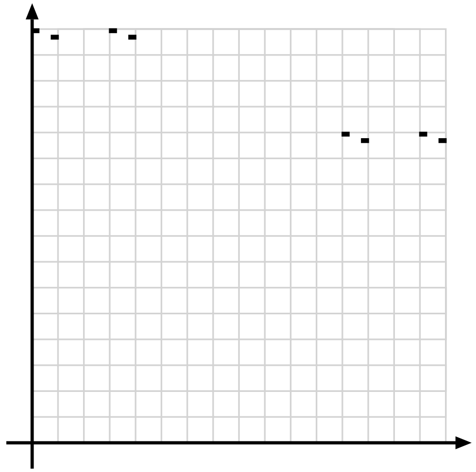

```
p(0).
p(s(X)) :- p(X).
p(X) :- not p(X).
```

```
p(s(s(s(s(s(s(s(s(s(s(0))))))))))) => 11
p(s(s(s(s(s(s(s(s(s(0)))))))))) => 10
p(s(s(s(s(s(s(s(s(0))))))))) => 9
p(s(s(s(s(s(s(s(0)))))))) => 8
p(s(s(s(s(s(s(0))))))) => 7
p(s(s(s(s(s(0)))))) => 6
p(s(s(s(s(0))))) => 5
p(s(s(s(0)))) => 4
p(s(s(0))) => 3
p(s(0)) => 2
p(0) => 1
```

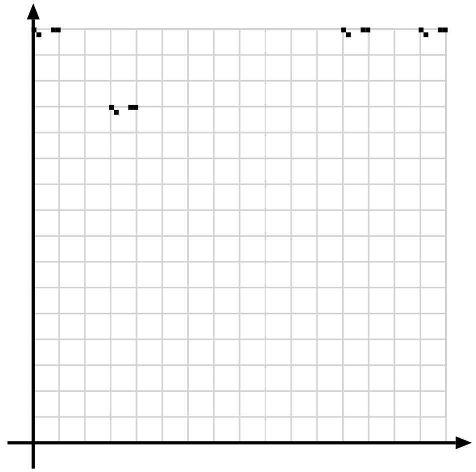
Examples of graphs of logic programs



```
n(0).
n(s(X))←n(X).
```

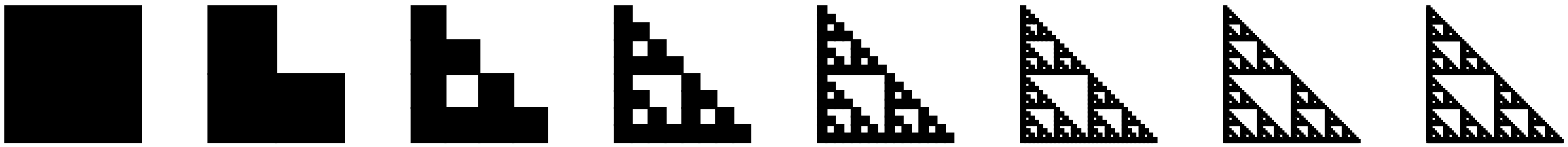


```
e(0).
e(s(X))←not e(X).
o(X)←not e(X).
```



```
p(0).
p(s(X))←p(X).
p(X)←not p(X).
```

(Hyperbolic) Iterated function systems (IFSs)



Space $\mathcal{H}$: Compact subsets of $\mathbb{R}^2$ with *Hausdorff metric*.

Set $\Omega = \{\omega_i\}$ of *contraction mappings* on $\mathbb{R}^2$.

$\bigcup \Omega(A) = \bigcup_i \omega_i(A)$ contraction on $\mathcal{H}$ with unique fixed point (*attractor*).

First representation theorem

P logic program. $R : I_P \to \mathbb{R}$ p-adic embedding.

$\left(\mathbb{R}^2, d, \Omega = \left\{\left(\omega_i^1, \omega_i^2\right)\right\}\right)$ hyperbolic IFS, attractor A.

Then

$$\mathrm{graph}(R(T_P)) = A$$

iff

$$\pi_1(A) = \mathrm{range}(R) \text{ and}$$
$$R(T_P)\left(\omega_i^1(a)\right) = \omega_i^2(a) \text{ for all } a \in \mathrm{graph}(R(T_P)) \text{ and all } i.$$

Second representation theorem

(Bader & Hitzler, Journal of Applied Logic 2004)

> P logic program with Lipschitz-continuous $R(T_P)$.
> Then there exists IFS with attractor graph$(R(T_P))$.

Idea: Set $\omega_i^2(x) = R(T_P)(\omega_i^1(x))$.
Choose $\omega_i^1(x)$ such that it generates range(R). This is possible with arbitrarily small contraction, the necessary size of which can be determined by the Lipschitz constant of $R(T_P)$.

Concrete approximation by interpolation

$a \in \mathbb{N}$ *accuracy.* (JAL to appear)

l injective level mapping (enumeration of B_P).

Interpolation points: $(R(I), R(T_P(I)))$, where $I \in D = \{A \mid l(A) < a\}$.

IFS with $\Omega_a = \{(\omega_i^1, \omega_i^2)\}$, where

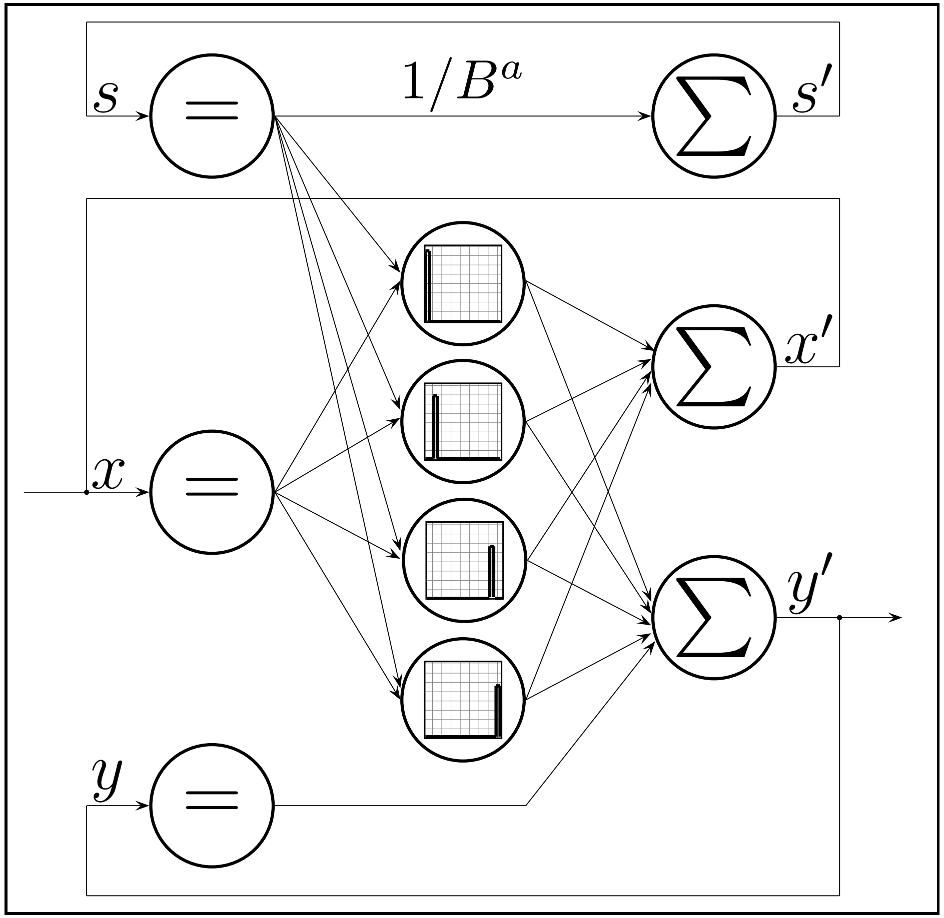
$$\omega_i^1(x) = \frac{1}{B^a}x + d_i^1$$

$$\omega_i^2(x) = \frac{1}{B^a} + R(T_P)\left(d_i^1\right) - \frac{R(T_P)(0)}{B^a}$$

Attractors A_a are graphs of continuous functions.

$(A_a)_a$ converges in function space (with sup-metric) to $R(T_P)$
if $R(T_P)$ Lipschitz-continuous.

Encoding as radial basis function network



Conclusions

The fixpoint completion is a stronger technical tool than previously noted.