Integrating Logic Programs and Connectionist Systems
A Constructive Approach

Sebastian Bader\textsuperscript{1}, Pascal Hitzler\textsuperscript{2}, Andreas Witzel\textsuperscript{3}

\textsuperscript{1}International Center for Computational Logic, Technische Universität Dresden, Germany
\textsuperscript{2}AIFB, Universität Karlsruhe, Germany
\textsuperscript{3}Department of Computer Science, Technische Universität Dresden, Germany

\textsuperscript{*}Sebastian Bader is supported by the GK334 of the German Research Foundation (DFG).
\textsuperscript{†}Pascal Hitzler is supported by the German Federal Ministry of Education and Research (BMBF) under the SmartWeb project, and by the European Union under the KnowledgeWeb Network of Excellence.
\textsuperscript{‡}Andreas Witzel is supported by Freunde und Förderer der Informatik an der TU Dresden e.V.
1. Motivation
2. Approximating Logic Programs
3. Multi-Layer Feed-Forward Networks
4. Radial Basis Function (RBF) Networks
5. Conclusions
Motivation

<table>
<thead>
<tr>
<th>Logic Programs (LP)</th>
<th>Connectionist Systems (CS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>well-defined semantics</td>
<td>robust</td>
</tr>
<tr>
<td>human-readable</td>
<td>adaptive</td>
</tr>
<tr>
<td>human-writable</td>
<td>trainable</td>
</tr>
</tbody>
</table>

Goal:
- Integrate both paradigms in order to exploit all advantages

One step towards achieving this goal:
- Transform LP into CS

What we have so far:
- Constructions for Propositional LP
- Non-constructive proofs for First-Order LP

In this work:
- Constructions for First-Order LP
A Simple Example

- **A Logic Program** $P$

  \[
  \begin{align*}
  \text{even}(0). & \quad \% \text{ 0 is an even number} \\
  \text{even}(s(X)) & \leftarrow \text{not even}(X). \quad \% \text{ the successor of a} \\
  & \quad \% \text{ non-even } X \text{ is even}
  \end{align*}
  \]

- **The Herbrand Base** $\mathcal{B}_P$ and some Interpretations

  \[
  \begin{align*}
  \mathcal{B}_P &= \{ \text{even}(0), \text{even}(s(0)), \text{even}(s^2(0)), \ldots \} \\
  I_1 &= \{ \text{even}(0), \text{even}(s(0)) \} \\
  I_2 &= \{ \text{even}(0), \text{even}(s^3(0)), \text{even}(s^4(0)), \text{even}(s^5(0)), \ldots \}
  \end{align*}
  \]

- **The Single-Step Operator** or **Meaning Function** $T_P$

  \[
  \begin{align*}
  I_1 & \xrightarrow{T_P} I_2 \quad \xrightarrow{T_P} \{ \text{even}(0), \text{even}(s^2(0)), \text{even}(s^3(0)) \} \\
  & \xrightarrow{T_P} \ldots \quad \xrightarrow{T_P} \{ \text{even}(0), \text{even}(s^2(0)), \text{even}(s^4(0)), \\
  & \quad \text{even}(s^6(0)), \text{even}(s^8(0)), \text{even}(s^{10}(0)), \ldots \}
  \end{align*}
  \]
Embedding $T_P$ in $\mathbb{R}$

- Enumerate $B_P$ using $\| \cdot \| : B_P \rightarrow \mathbb{N} \setminus \{0\}$
  \[
  \|even(s^n(0))\| := n + 1
  \]

- Embed $I \in J_P$ into $\mathbb{R}$ using $R(I) := \sum_{A \in I} 3^{-\|A\|}$
  \[
  R\left(\{even(0), even(s^2(0))\}\right) = 0.1010000\ldots_3
  \]

- Embed $T_P$ into $\mathbb{R}$:
  \[
  I \in J_P \xrightarrow{T_P} I' \in J_P
  \]
  \[
  \begin{array}{c}
  R^{-1} \\
  f_P
  \end{array}
  \]
  \[
  x \in D_f \xrightarrow{f_P} x' \in D_f
  \]

where $D_f := \{R(I) \mid I \in J_P\}$
In general, the graph is more complicated and not on a straight line!
Goal: approximate $f_P$ (the embedded $T_P$) up to $\varepsilon$

Consider $x, x' \in D_f$:

\[
\begin{align*}
  x &= 0.00101011010000000\ldots_3 \\
      &= 0.000000\ldots_3 \\
  x' &= 0.0010101101011111\ldots_3
\end{align*}
\]

Maximum difference $\delta_l := \sum_{i>l} 3^{-i} = \frac{1}{3^l \cdot 2}$

Greatest relevant output level $o_\varepsilon := \min \{ n \in \mathbb{N} | \delta_n < \varepsilon \}$

Assume $T_{P'}$ and $T_P$ agree on all atoms of level $\leq o_\varepsilon$

$\Rightarrow$ $f_{P'}$ and $f_P$ agree on the first $o_\varepsilon$ digits

$\Rightarrow$ $f_{P'}$ approximates $f_P$ up to $\varepsilon$
The Instance of $P$ up to $o_\varepsilon$

- **Goal**: find $P'$ such that $T_{P'}$ and $T_P$ agree on atoms of level $\leq o_\varepsilon$
- Inclusion of $A$ in $T_P(I)$ depends only on clauses with head $A$
- $P' := \{ A \leftarrow B \in \mathcal{G}(P) \mid \|A\| \leq o_\varepsilon \}$
  where $\mathcal{G}(P) := \text{set of all ground instances of clauses from } P$
- $P'$ is finite if $P$ is covered, i.e. if there are no local variables
- Greatest relevant input level
  \[ \hat{o}_\varepsilon := \max \{ \|L\| \mid L \text{ is body literal of some clause in } P' \} \]
- $T_{P'}$ depends only on atoms of level $\leq \hat{o}_\varepsilon$
  \[ \Rightarrow f_{P'} \text{ depends only on the first } \hat{o}_\varepsilon \text{ digits} \]
  \[ \Rightarrow f_{P'} \text{ is constant for all inputs which agree on first } \hat{o}_\varepsilon \text{ digits} \]
  \[ \Rightarrow f_{P'} \text{ consists of finitely many constant pieces} \]
Our Example with $\varepsilon = 0.02$

\[ f_{P'}(R(I)) \]

- $\omega_{0.02} = 3$
- $P' = e(0)$.
- $e(s(0)) \leftarrow \neg e(0)$.
- $e(s^2(0)) \leftarrow \neg e(s(0))$.

$\Rightarrow \hat{\omega}_{0.02} = 2$
Building a CS with Step Activation Functions

\[ f_{P^r}(R(I)) \]

\[ R(\text{I}) \]

\[ \cdot \cdot \cdot 0.00000 \quad \cdot \cdot \cdot 0.00111 \quad \cdot \cdot \cdot 0.01000 \quad \cdot \cdot \cdot 0.01111 \quad \cdot \cdot \cdot 0.10000 \quad \cdot \cdot \cdot 0.10111 \quad \cdot \cdot \cdot 0.11000 \quad \cdot \cdot \cdot 0.11111 \quad \cdot \cdot \cdot \]
Each \( \square \) computes
- 0, if weighted sum of inputs \( \leq 0 \)
- 1, otherwise
Building a CS with Sigmoidal Activation Functions

Approximate the step functions

Divide $\varepsilon$ into $\varepsilon'$ for $P'$ and $\varepsilon''$ for the sigmoidals. The closest constant piece yields the zoom-out factor.
Approximate the step functions by sigmoidals.
Approximate the step functions by sigmoidals.
Building a CS with Sigmoidal Activation Functions

Approximate the step functions by sigmoidals

Divide $\varepsilon$ into $\varepsilon'$ for $P'$ and $\varepsilon''$ for the sigmoidals

The closest constant piece yields the zoom-out factor
Describe each constant piece by two triangles or raised cosines:
Refining an Existing Network

- Decreasing $\varepsilon$ will only add clauses to $P'$
- Consequence:
  - Constant pieces may be divided into smaller pieces
  - Some parts may be raised
- For $\varepsilon = 0.007$, we get:

![Graph showing the effects of decreasing $\varepsilon$.]
Conclusions and Problems

What we had before:
- Methods to construct CS for propositional LP
- Non-constructive proofs for the existence of CS approximating first-order LP

New results:
- Methods for constructing CS approximating first-order LP
- Method for iterative refinement

Problem:
- Floating point precision in real computers is very limited, so we can represent only few atoms

Possible remedy:
- Distribute representation on several input/output nodes
Conclusions and Problems

What we had before:
- Methods to construct CS for propositional LP
- Non-constructive proofs for the existence of CS approximating first-order LP

New results:
- Methods for constructing CS approximating first-order LP
- Method for iterative refinement

Problem:
- Floating point precision in real computers is very limited, so we can represent only few atoms

Possible remedy:
- Distribute representation on several input/output nodes

Thank you for your attention.