The First Order CORE Method

Steffen Hölldobler
International Center for Computational Logic
Technische Universität Dresden
Germany

- The Core Method
- Logic Programs
- Mapping Interpretations to Real Numbers
- Approximation of Interpretations
- Constructive Approaches
- Implementation
- Open Problems

"Logic is everywhere..."
The CORE Method

- Relate logic programs and connectionist systems.
- Embed interpretations into (vectors of) real numbers.
- Hence, obtain an embedded version of the $T_P$-operator.
- Construct a network computing one application of $f_P$.
- Add recurrent connections from output to input layer.
- Compute (or approximate) the least fixed point of $T_P$. 

**Diagram:**

```
Symbolic System -- writable
                    | embedding
                    | readable
                    v
Connectionist System -- trainable
```

**Symbols:**

- $T_P$: $T_P$-operator
- $f_P$: $f_P$-function

Steffen Hölldobler
The First Order CORE Method
Logic Programs

- A logic program $\mathcal{P}$ over a first-order language $\mathcal{L}$ is a finite set of clauses

$$A \leftarrow L_1 \land \ldots \land L_n,$$

where $A$ is an atom, $L_i$ are literals and $n \geq 0$.

- $T_{\mathcal{L}}$ is the set of all ground terms over $\mathcal{L}$.

- $B_{\mathcal{L}}$ is the set of all ground atoms over $\mathcal{L}$ called Herbrand base.

- A Herbrand interpretation $I$ is a mapping $B_{\mathcal{L}} \rightarrow \{\top, \bot\}$.

- $2^{B_{\mathcal{L}}}$ is the set of all Herbrand interpretations.

- $g\mathcal{P}$ is the set of all ground instances of clauses in $\mathcal{P}$.

- Immediate consequence operator $T_{\mathcal{P}} : 2^{B_{\mathcal{L}}} \rightarrow 2^{B_{\mathcal{L}}}$:

$$T_{\mathcal{P}}(I) = \{A \mid \text{there is a clause } A \leftarrow L_1 \land \ldots \land L_n \in g\mathcal{P} \text{ such that } I \models L_1 \land \ldots \land L_n\}.$$ 

- $I$ is a supported model iff $T_{\mathcal{P}}(I) = I$. 
Two Examples

▶ Natural numbers

\[
\begin{align*}
n0 & \quad \% \ 0 \ is \ a \ natural \ number. \\
s_nX & \leftarrow nX \quad \% \ The \ successor \ sX \ is \ a \ natural \\
& \quad \% \ number \ if \ X \ is \ a \ natural \ number.
\end{align*}
\]

▶ Even and odd numbers

\[
\begin{align*}
e0 & \quad \% \ 0 \ is \ an \ even \ number. \\
es_nX & \leftarrow oX \quad \% \ The \ successor \ of \ an \ odd \ X \ is \ even. \\
oX & \leftarrow \neg eX \quad \% \ If \ X \ is \ not \ even \ then \ it \ is \ odd.
\end{align*}
\]

▶ Herbrand base

\[
B_{\mathcal{L}} = \{e0, es0, \ldots, o0, os0, \ldots\}
\]

▶ Some interpretations

\[
\begin{align*}
I_1 &= \{es^{2m}0 \mid m \geq 1\} & I_3 &= I_1 \cap I_2 = \emptyset \\
I_2 &= \{os^{2m+1}0 \mid m \geq 0\} & I_4 &= I_1 \cup I_3
\end{align*}
\]
The Immediate Consequence Operator

\( \mathcal{I}_P(I) = \{ A \mid \text{there is a clause } A \leftarrow L_1 \land \ldots \land L_n \in g \mathcal{P} \) such that \( I \models L_1 \land \ldots \land L_n \} \).

- **Natural Numbers** \( \{n0, nsX \leftarrow nX\} \)

\[
\begin{align*}
\emptyset & \mapsto \{n0\} \\
\{n0\} & \mapsto \{n0, ns0\} \\
\{n0, ns0\} & \mapsto \{n0, ns0, nss0\} \\
B & \mapsto B
\end{align*}
\]

- **Even and odd numbers** \( \{e0, esX \leftarrow oX, oX \leftarrow \neg eX\} \)

\[
\begin{align*}
\emptyset & \mapsto \{e0, oX \mid X \in T\} \\
\{oX \mid X \in T\} & \mapsto \{e0, esX, oX \mid X \in T\} \\
\{es^{2m}0 \mid n \geq 0\} & \mapsto \{e0, os^{2m+1}0 \mid m \geq 0\} \\
\{os^{2m+1}0 \mid n \geq 0\} & \mapsto \{e0, es^{2m}0 \mid m \geq 0\} \\
B & \mapsto \{e0, esX \mid X \in T\}
\end{align*}
\]
The Initial Approach

- Hölldobler, Kalinke, Störr 1999:
  Can the core method be extended to first-order logic programs?

- Problem
  - Given a logic program $\mathcal{P}$ over a first order language $\mathcal{L}$
    together with $\top_{\mathcal{P}} : 2^{B_{\mathcal{L}}} \rightarrow 2^{B_{\mathcal{L}}}$.
  - $B_{\mathcal{L}}$ is countably infinite.
  - The method used to relate propositional logic and connectionist systems
    is not applicable.
  - How can the gap between the discrete, symbolic setting of logic
    and the continuous, real valued setting of connectionist networks be closed?
The Goal

Find $\text{rep} : 2^B \mathcal{L} \rightarrow \mathbb{R}$ and $f_P : \mathbb{R} \rightarrow \mathbb{R}$ such that the following conditions hold.

- $\mathcal{T}_P(I) = I'$ implies $f_P(\text{rep}(I)) = \text{rep}(I')$.
- $f_P(x) = x'$ implies $\mathcal{T}_P(\text{rep}^{-1}(x)) = \text{rep}^{-1}(x')$.

- $f_P$ is a sound and complete encoding of $\mathcal{T}_P$.
- $\mathcal{T}_P$ is a contraction on $2^B \mathcal{L}$ iff $f_P$ is a contraction on $\mathbb{R}$.
- The contraction property and fixed points are preserved.
- $f_P$ is continuous on $\mathbb{R}$.

- A connectionist network approximating $f_P$ is known to exist.
- $f_P$ and, hence, $\mathcal{T}_P$ can be trained by backpropagation and related training methods.
Level Mappings

- Let $\mathcal{P}$ be a program over a first order language $\mathcal{L}$.
- A level mapping for $\mathcal{P}$ is a function $l : B_{\mathcal{L}} \rightarrow \mathbb{N}$.
  - We define $l(\neg A) = l(A)$.
- Examples
  - Natural Numbers $\{n0, nsX \leftarrow nX\}$
    $$l(ns^m0) = m + 1$$
  - Even and odd numbers $\{e0, esX \leftarrow oX, oX \leftarrow \neg eX\}$
    $$l(es^m0) = 2n + 1, \quad l(os^m0) = 2m + 2$$
Acyclic Logic Programs

We can associate a metric $d_L$ with $L$ and a level mapping $l$ as follows. Let $I, J \in 2^B_L$:

$$d_L(I, J) = \begin{cases} 0 & \text{if } I = J \\ 2^{-n} & \text{if } n \text{ is the smallest level on which } I \text{ and } J \text{ differ.} \end{cases}$$

Proposition (Fitting 1994) $(2^B_L, d_L)$ is a complete metric space.

$P$ is said to be acyclic wrt a level mapping $l$, if for every $A \leftarrow L_1 \land \ldots \land L_n \in g\ P$ we find $l(A) > l(L_i)$ for all $i$.

$P$ is said to be acyclic if $P$ is acyclic wrt some level mapping.

Both running examples are acyclic.

Proposition  Let $P$ be an acyclic logic program wrt $l$ and $d_L$ the metric associated with $L$ and $l$, then $T_P$ is a contraction on $(2^B_L, d_L)$. 

Steffen Hölldobler
The First Order CORE Method
Mapping Interpretations to Real Numbers

- Let $l$ be a bijective level mapping.
- Let $\text{rep}$ be defined as
  \[ \text{rep}(I) = \sum_{A \in I} 4^{-l(A)}. \]
- Example

\[
\begin{array}{c}
B_L = \{ e0, o0, es0, os0, ess0, \ldots \} \\
\text{rep}(\{e0\}) = 0.10000000_4 = 0.25_{10} \\
\text{rep}(\{e0, es0, ess0\}) = 0.10101010_4 \approx 0.27_{10}
\end{array}
\]
The Set of all Embedded Interpretations

- Let $\mathcal{E} = \{\text{rep}(I) \mid I \in 2^{B\mathcal{L}}\}$

- Bader 2003 $\mathcal{E}$ is the attractor of an iterated function system.

- Proposition $\text{rep}$ is a bijection between $2^{B\mathcal{L}}$ and $\mathcal{E}$.

  We have a sound and complete encoding of interpretations.

- Proposition $\mathcal{E}$ is compact.
Mapping Immediate Consequence Operators to Functions on the Reals

We define \( f_P : \mathcal{E} \rightarrow \mathcal{E} : r \mapsto \text{rep}(T_P(\text{rep}^{-1}(r))). \)

We have a sound and complete encoding of \( T_P \).

**Proposition** Let \( \mathcal{P} \) be an acyclic program wrt a bijective level mapping. \( f_P \) is a contraction on \( \mathcal{E} \).

Contraction property and fixed points are preserved.
Approximating Continuous Functions

- **Corollary** \( f_P \) is continuous.

- **Recall Funahashi’s theorem:**

  - Let \( K \subseteq \mathbb{R}^n \) be compact.
    Every continuous function \( f : K \to \mathbb{R} \) can be uniformly approximated by input-output functions of 3-layer feed forward networks.

- **Theorem** \( f_P \) can be uniformly approximated by input-output functions of 3-layer feed-forward networks.

  - \( T_P \) can be approximated as well by applying \( \text{rep}^{-1} \).

  Connectionist network approximating immediate consequence operator exists.
An Example

Consider $\mathcal{P} = \{n0, \ nsX \leftarrow nX\}$ and let $|ns^m0| = m + 1$.

- $\mathcal{P}$ is acyclic wrt $|\cdot|$, $|\cdot|$ is bijective, $\text{rep}(B_L) = \frac{1}{3}$.
- $f_{\mathcal{P}}(\text{rep}(I)) = 4^{-|n0|} + \sum_{nX \in I} 4^{-|nsX|} = 4^{-|n0|} + \sum_{nX \in I} 4^{-(|nX|+1)} = \frac{1 + \text{rep}(I)}{4}$.

Approximation of $f_{\mathcal{P}}$ to accuracy $\varepsilon$ yields

$$\tilde{f}(x) \in \left[\frac{1 + x}{4} - \varepsilon, \frac{1 + x}{4} + \varepsilon\right].$$

Starting with some $x$ and iterating $\tilde{f}$ yields in the limit a value

$$\overline{r} \in \left[\frac{1 - 4\varepsilon}{3}, \frac{1 + 4\varepsilon}{3}\right].$$

Select $r \in \mathcal{E}$ with minimal distance to $\overline{r}$.

Applying $\text{rep}^{-1}$ to $r$ we find

$$ns^m0 \in \text{rep}^{-1}(r) \text{ if } m < -\log_4 \varepsilon - 1.$$
Approximation of Interpretations

- Let \( \mathcal{P} \) be a logic program over a first order language \( \mathcal{L} \) and \( \text{l} \) a level mapping.
- An interpretation \( I \) approximates an interpretation \( J \) to a degree \( n \in \mathbb{N} \) if for all atoms \( A \in \mathcal{B}_\mathcal{L} \) with \( \text{l}(A) < n \) we find \( I(A) = \top \) iff \( J(A) = \top \).
- \( I \) approximates \( J \) to a degree \( n \) iff \( d_\mathcal{L}(I, J) \leq 2^{-n} \).
Approximation of Supported Models

- Given an acyclic logic program $\mathcal{P}$ with bijective level mapping.
- Let $T_\mathcal{P}$ be the immediate consequence operator associated with $\mathcal{P}$ and $M_\mathcal{P}$ the least supported model of $\mathcal{P}$.
- We can approximate $T_\mathcal{P}$ by a 3-layer feed-forward network.
- We can turn this network into a recurrent one.

Does the recurrent network approximate the supported model of $\mathcal{P}$?

**Theorem** For an arbitrary $m \in \mathbb{N}$ there exists a recursive network with sigmoidal activation function for the hidden layer units and linear activation functions for the input and output layer units computing a function $\tilde{f}_\mathcal{P}$ such that there exists an $n_0 \in \mathbb{N}$ such that for all $n \geq n_0$ and for all $x \in [-1, 1]$ we find

$$d_L(\text{rep}^{-1}(\tilde{f}_\mathcal{P}^n(x)), M_\mathcal{P}) \leq 2^{-m}.$$
First Order Core Method – Extensions

- Detailed study in (topological) continuity of semantic operators
  Hitzler, Seda 2003 and Hitzler, Hölldobler, Seda 2004:
  - many-valued logics,
  - larger class of logic programs,
  - other approximation theorems.
- A core method for reflexive reasoning
  Hölldobler, Kalinke, Wunderlich 2000.
- The graph of $f_P$ is an attractor of some iterated function system
  Bader 2003 and Bader, Hitzler 2004:
  - representation theorems,
  - fractal interpolation,
  - core with units computing radial basis functions.
Constructive Approaches: Approximating Piecewise Constant Functions

Consider graph of $f_P$. Approximate $f_P$ up to a given level $l$. Construct network computing piecewise constant function.

Step activation functions. Sigmoidal activation functions. Radial basis functions.
Constructive Approaches: Approximating Piecewise Constant Functions

- Consider graph of $f_P$.
- Approximate $f_P$ up to a given level $l$.

Construct core computing piecewise constant function.

Step activation functions.
Sigmoidal activation functions.
Radial basis functions.
Constructive Approaches: Approximating Piecewise Constant Functions

- Consider graph of $f_P$.
- Approximate $f_P$ up to a given level $l$.
- Construct core computing piecewise constant function.
  - Step activation functions.

Sigmoidal activation functions.
Radial basis functions.
Constructive Approaches: Approximating Piecewise Constant Functions

- Consider graph of $f_P$.
- Approximate $f_P$ up to a given level $l$.
- Construct core computing piecewise constant function.
  - Step activation functions.
  - Sigmoidal activation functions.
  - Radial basis functions.
Constructive Approaches: Approximating Piecewise Constant Functions

- Consider graph of $f_P$.
- Approximate $f_P$ up to a given level $l$.
- Construct core computing piecewise constant function.
  - Step activation functions.
  - Sigmoidal activation functions.
  - Radial basis functions.

A Problem

- The accuracy of this approach is very limited.
- E.g., on a 32 bit computer only 16 atoms can be represented.
- We need to use real vectors instead of a single real number to represent interpretations.
Multi-Dimensional Level Mappings

▶ An $k$-dimensional level mapping $| \cdot |$ assigns to each ground atom a level $l \in \mathbb{N}^+$ and a dimension $d \in \{1, \ldots, k\}$:

▶ Example

$$|es^m0| = (m + 1, 1), \quad |os^m0| = (m + 1, 2).$$

▶ We can now define the embedding $\text{rep}$ as follows:

$$\text{rep}(I) = \sum_{A \in I} \text{rep}(A)$$

where

$$\text{rep}(A) = (\text{rep}_1(A), \ldots, \text{rep}_k(A))$$

and

$$\text{rep}_j(A) = \begin{cases} 4^{-l(A)} & \text{if } |A| = (l, j) \\ 0 & \text{otherwise} \end{cases}$$

▶ All results can be extended to $k$-dimensional level mappings.
Implementation

- A first prototype (FineBlend) was implemented in Witzel 2006.
  - Merging of the techniques described above and supervised growing neural gas (SGNG) developed in Fritzke 1998.
  - Radial basis function network approximating $T_P$.
  - Very robust with respect to noise and damage.
  - Trainable using a version of backpropagation together with techniques from SGNG.
FineBlend versus SGNG
FineBlend: Unit Failure

![Error vs #units plot](image)
FineBlend: Iterating Random Inputs

![Graph showing FineBlend: Iterating Random Inputs](image_url)
Open Problems

► How can first order terms be represented and manipulated in a connectionist system (Pollack 1990, Hölldobler 1990, Plate 1991)?

► Can the mapping rep be learned (Gust, Kühnberger 2005)?

► How can first order rules be extracted from a connectionist system?

► How can multiple instances of first order rules be represented in a connectionist system (Shastri 1990)?

► What does a theory for the integration of logic and connectionist systems look like?

► Can such a theory be applied in real domains outperforming conventional approaches?