Some Remarks on the Definability of Transitive Closure in First-order Logic and Datalog

Uwe Keller

June 1, 2004

Abstract

In the last WSML phone conference we had a brief discussion about the expressivity of First-order Logic and Datalog resp. the relation between the expressiveness of those two languages.

In particular, there has been some confusion around the description of the transitive closure R^+ of some binary relation R.

In this short document, we want to clarify the situation and hope to remedy the confusion.

1 Starting point

During the discussion in the last WSML phone conference the statement arose that Datalog with it's particular semantics can express some things which can not be expressed in the First-order Logic (FOL) under the standard model-theoretic (resp. Tarski) semantics [14].

As an example the *transitive closure* of a (binary) relation has been mentioned. Some people didn't believe this claim because it's a straightforward task to describe transitive closure in Datalog and it's been not obvious why this can not be done in FOL.

We will clarify the situation in this document by (a) showing that transitive closure can not be expressed in a First-order logic, (b) showing that it can be expressed in Datalog, (c) discussing the precise reason for the difference and (d) giving several references to interesting related work which sheds some light on the relation of First-order Logic, Datalog, transitive closure and inductive definitions. Moreover, we will present some logics extending FOL in which the transitive closure of a binary relation is definable very easily.

In a nutshell, we will show that the very reason for the definability of transitive closure in Datalog is the so-called Closed-World-Assumption (CWA).

Since we want to define the transitive closure operator in some logical languages, we first have to have a common understanding of what this operator means. Thus, we start with the definition of the transitive closure operator.

Definition 1.1 (Transitive closure). Let U be some set and $R \subseteq U \times U$ be a binary relation on the universe U. Then the *transitive closure* T_R of relation R is the smallest relation $T_R \subseteq U \times U$ with

1. $R \subseteq T_R$ (i.e. for all $e_1, e_2 \in U$ it holds that if $R(e_1, e_2)$ then $T_R(e_1, e_2)$) *and*
2. T_R is transitive (that is, for all $e_1, e_2, e_3 \in U$ it holds that if $T_R(e_1, e_2)$ and $T_R(e_2, e_3)$ then $T_R(e_1, e_3)$)

The transitive closure of relation R is commonly denoted by R^+.

Another common characterization of the transitive closure operator (in an algebraic way) is as follows:

Definition 1.2 (Transitive closure). Let U be some set and $R \subseteq U \times U$ be a binary relation on the universe U. Then the *transitive closure* R^+ of relation is defined by

$$R^+ = \bigcup_{n \in \mathbb{N}} R^n$$

where the n-th power R^n of a relation R is defined by

$$R^n = \begin{cases} Id_U & \text{if } n = 0 \\ R^{n-1} \circ R & \text{if } n > 0 \end{cases}$$

and

$$Id_U = \{(e, e) : e \in U\}$$

In fact, it's not hard to show that both definitions coincide.

2 Datalog and transitive closure

Now, if we look at definition 1.1 then we can immediately derive an algorithm for computing R^+ for a given relation R: We start with the relation R as our initial relation R^+ and add an additional tupel (e_1, e_3) to R^+ for all $e_1, e_2, e_3 \in U$ where $R^+(e_1, e_2)$ and $R^+(e_2, e_3)$. That means, in each step the relation R^+ becomes "more transitive". We stop as soon as the constructed relation R^+ is indeed transitive. For finite relations R the algorithm is garantueed to terminate.

This algorithm can be easily encoded in a Datalog program P:

```
trans-closure-r(x,y) <- r(x,y)                                        [R1]
trans-closure-r(x,z) <- trans-closure-r(x,y), trans-closure-r(y,z)    [R2]
```

If we assume that additionally the relation R is represented as a set of ground facts (in addition `[R1]` and `[R2]` in the program P) then the minimal model M_P of P can be computed as the least fixpoint of the immediate consequence operator T_P.

In this minimal model M_P the interpretation of `trans-closure-r` is according to `[R1]` a superset of R and according to `[R2]` a transitive relation. Since M_P is computed as the *least* fix point and a fixpoint of the operator T_p represents a model we can conclude that the interpretation of `trans-closure-r` satisfies definition 1.1 and thus coincides with the transitive closure R^+ of the relation R (which is the interpretation of predicate `r` in M_P).

In this respect, we can define resp. represent the transitive closure of some relation in Datalog by means of a program similar to program P.

3 First-order Logic and transitive closure

Let's briefly reconsider the Datalog program P given above. Since the program is consituted by simple clauses (over some signature Σ), it represents a set of Σ-formulas in a FOL. Thus it seems to be a bit strange in the first place that the same argument as given in section 2 should not be applicable here as well.

Thus, in this section we will indeed show that the transitive closure can not be described in a FOL.

Naturally, in order to show that something (for instance a certain class of structures) can not be defined in a First-order Logic, we first have to clarify, what we understand by the notion of *definability* in the context of a logic.

Definition 3.1 (Definability of a class of structures in FOL). Let Σ be some signature and K a class of Σ-structures (that is structures which can be used to interpret any formula over signature Σ).

We say that the class K is *definable* (over signature Σ) if there is a closed Σ-formula ϕ such that for every Σ-structure $\mathcal{S}$

$$\mathcal{S} \models \phi \qquad \textit{iff} \qquad \mathcal{S} \in K$$

We write $Mod_\Sigma(\phi)$ for the class of Σ-structures $\mathcal{S}$ such that $\mathcal{S} \models \phi$ holds (i.e. the class of models of ϕ). In other words, a class K of structures is definable in a FOL iff there is a closed formula ϕ with $Mod_\Sigma(\phi) = K$.

Let Σ be a signature that contains at least the predicates R and R^+. According to this definition, we are interested in the definability of a particular class K^{TCL} of Σ-structures in which the predicate R^+ is acutally interpreted as the transitive closure of the relation corresponding to predicate R, that is

$$K^{TCL} = \{\mathcal{S} \,:\, \mathcal{S} = (D^{\mathcal{S}}, I^{\mathcal{S}})\ \Sigma - \textit{structure}\,,\, I^{\mathcal{S}}(R^+) = (I^{\mathcal{S}}(R))^+\}$$

Now, the question is: Is K^{TCL} definable (over Σ)? Is there a sentence ϕ^{TCL} such that $K^{TCL} = Mod_\Sigma(\phi^{TCL})$? If this would be the case, then we could always use the transitive closure for a relation described by a predicate R in FOL descriptions by introducing some new symbol R^+ and fix the meaning of this new symbol by using the defining sentence ϕ^{TCL}.

Unfortunately, we will prove that this is not the case.

Theorem 3.2 (Undefinability of K^{TCL} in FOL). *Let Σ be a signature with $\{R, R^+\} \subseteq \Sigma$. The class K^{TCL} of Σ-structures is not Σ-definable in a First-order Logic $\mathcal{L}$. In other words, the transitive closure operator can not be precisely charcterized within the framework of FOL.*

Before we give the proof of theorem 3.2, we first want to mention one of the important properties of First-order logics: The so-called *compactness* of FOL.

Theorem 3.3 (Compactness of FOL). *Let Φ be a (possibly infinite) set of formulas of a First-order $\mathcal{L}$. Then Φ is satisfiable (resp. has a model) iff. every finite $\Phi_0 \subset \Phi$ is satisfiable.*

In fact, this particular property distinguishes FOL from other, more expressive logics like Second-order logics $\mathcal{L}_{II}$ or infinitary logics like $\mathcal{L}_{\omega_1\omega}$. More precisely, Lindström [11] has shown that First-order logics are the *most expressive logics* in which both the compactness theorem as well as the Löwenheim-Skolem-theorem hold.

Surprisingly, we searched the Internet for a (model-theoretic) proof of theorem 3.2, but we have not been successful in this respect. Moreover, several textbooks on FOL [8] and model theory [7, 6] didn't give a proof of our claim. Thus, we decided to come up with a direct (model-theoretic) proof ourselves. Fortunately, it was possible to adapt a standard approach to proving undefinability of a certain class of structures by using the *compactness property* of First-order Logics. Thus, a proof was not really hard to construct.

Proof of theorem 3.2. The main line of argument of the proof is as follows: We assume that the contrary, that means that K^{TCL} is Σ-definable in $\mathcal{L}$. Then we consider particular models $\mathcal{S}_n$ of ϕ^{TCL} which additionally have some particular property ψ_n which is related to property of representing the transitive closure. The key point is that for each $n_0 \in \mathbb{N}$ we can find a model $\mathcal{S}_n, n_0 \leq n$ for $\{\phi^{TCL}\} \cup \{\psi_i : 1 \leq i \leq n_0\}$ but the (infinite) set $\overline{K^{TCL}} = \bigcup_{n\in\mathbb{N}}\{\psi_n\}$ of formulas contradicts the property ϕ^{TCL}. After we have succeeded with this construction, we can immediately exploit the compactness theorem and derive a contradication, since every finite subset of $\Phi^* = \{\phi^{TCL}\} \cup \overline{K^{TCL}}$ is satisfiable whereas Φ^* is not satisfiable.

We start with the construction of $\overline{K^{TCL}}$. Since it is possible to define properties (1) and (2) in the definition 1.1 in FOL, the set $\overline{K^{TCL}}$ has to refer to the minimality of the relation R^+ representing the transitive closure of some relation R.

Consider some Σ-structure $\mathcal{S} = (U^{\mathcal{S}}, I^{\mathcal{S}})$. Informally speaking, the relation $I^{\mathcal{S}}(R^+)$ satisfying properties (1) and (2) in definition 1.1 is not minimal, if there is an arc $(e, e') \in I^{\mathcal{S}}(R^+)$ such that there is no evidence for this arc in the corresponding relation $I(R)$, that is there is no R-path (of some length) connecting the elements $e \in U^{\mathcal{S}}$ and $e' \in U^{\mathcal{S}}$. Thus, for constructing the set $\overline{K^{TCL}}$ which contradicts ϕ^{TCL} we can use the following statement for ψ_n ($n \in \mathbb{N}$): *$I^{\mathcal{S}}(R^+)$ is not minimal because of a R-path of length $\leq n$*. Clearly, the set $\overline{K^{TCL}} = \bigcup_{1 \leq n} \{\psi_n\}$ expresses that $I^{\mathcal{S}}(R^+)$ is *not* minimal and thus it is *not* the transitive closure of relation R. Hence $\overline{K^{TCL}}$ contradicts ϕ^{TCL}.

More precisely, our informal statement can be formalized by means of the following closed Σ-formula ψ_n in $\mathcal{L}$:

$$\psi_n \equiv \exists e, e' : (R^+(e, e') \wedge \neg(\exists e_0, \ldots, e_n : e_0 \doteq e \wedge e_n \doteq e' \wedge \bigwedge_{1 \leq i \leq n} R(e_{i-1}, e_i)))$$

Please note that the equality symbol is not necessary for the proof; we can easily give an alternative formulation for ψ_n which does not use the equality symbol at all!

Next, we present the particular class of Σ-structures $\mathcal{S}_n = (U^{\mathcal{S}_n}, I^{\mathcal{S}_n})$ that we consider in the proof.

Let $G_n = (U_n, R_n)$ be a graph with nodes $U_n = \{0, 1, \ldots, n\} \subseteq \mathbb{N}_0$ and edges $R_n = \{(i, i+1) : 0 \leq i \leq n-1\} \subseteq U_n \times U_n$. The i-th power of graph G_n is defined as $G_n^i = (U_n, R_n^i)$ and the transitive closure of G_n is defined as $G_n^+ = (U_n, R_n^+)$. The different graphs are shown in figure 1. It's easy to see, that the transitive closure G_n^+ of G_n coincides with the union of the first n i-powers of G_n, that is $G_n^+ = \bigcup_{1 \leq i \leq n} G_n^i$, since the longest path in G_n is the path connecting 0 and n of length n. Furthermore, for every $i < j$ the graphs G_j^+ is a proper extension (resp. supergraph) of graph G_i^+.

The Σ-structure $\mathcal{S}_n$ that we use later represent the graph G_n as well as it's transitive closure G_n^+ as the interpretation of the predicates $R \in \Sigma$ and $R^+ \in \Sigma$. That means, $U^{\mathcal{S}_n} = U_n$, $I^{\mathcal{S}_n}(R) = R_n$ and $I^{\mathcal{S}_n}(R^+) = R_n^+$.

Now, we glue the single pieces of the puzzle together: We assume that K^{TCL} is Σ-definable in FOL. According to the definition of the notion of definability that means that there is a closed Σ-formula ϕ^{TCL} in $\mathcal{L}$ such that $K^{TCL} = Mod_{\Sigma}(\phi^{TCL})$. Clearly, the structures $\mathcal{S}_n$ from above are models of ϕ^{TCL}, that is, $\{\mathcal{S}_n : n \in \mathbb{N}_0\} \subseteq K^{TCL}$.

Let $\overline{K^{TCL}}$ denote the following infinite set of sentences

$$\overline{K^{TCL}} = \bigcup_{n \in \mathbb{N}} \{\psi_n\}$$

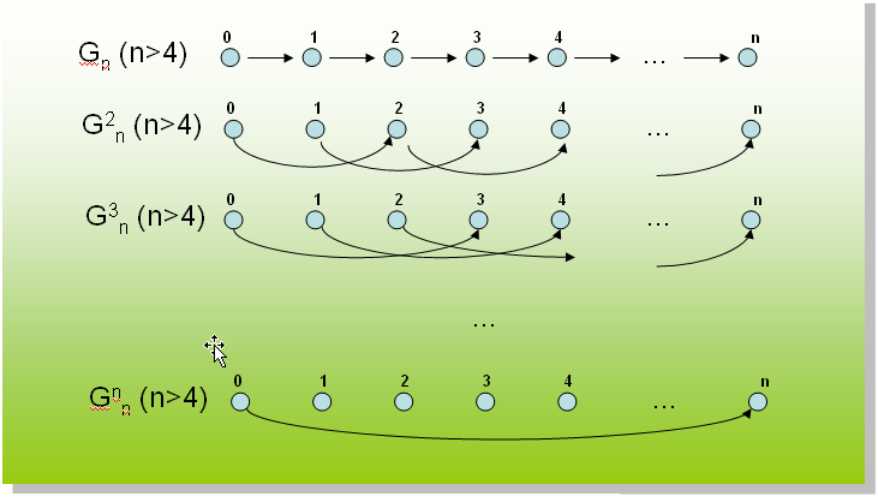

Figure 1: The graph G_n and it's i-th powers G_n^i

and let Φ^* be the (infinite) set

$$\Phi^* = \{\phi^{TCL}\} \cup \overline{K^{TCL}} = \{\phi^{TCL}\} \cup \bigcup_{n\in\mathbb{N}} \{\psi_n\}$$

of closed Σ-formulas.

Consider an arbitrary *finite* subset $\Phi \subset \Phi^*$. Then, there is a $n_0 \in \mathbb{N}$ such that $\Phi \subseteq \Phi_{n_0} = \{\phi^{TCL}\} \cup \bigcup_{n\in\mathbb{N}, n\leq n_0} \{\psi_n\} \subset \Phi^*$. Please note, that Φ_{n_0} is a finite set of closed Σ-formulas as well. If we can show that Φ_{n_0} is satisfiable by some structure $\mathcal{S}$ then clearly $\mathcal{S} \models \Phi$ holds too. Since Φ is an arbitrary finite subset of Φ^*, we can use the *compactness theorem* (theorem 3.3) to conclude that Φ^* itself (that means an infinite set of Σ-sentences) is satisfiable.

To do so, we consider the set Φ_{n_0} and give a model for this set of formulas. In particular, we have to look at each formula ϕ in Φ_{n_0}. Consider the structures $\mathcal{S}_n$ defined above.

- If $\phi = \phi^{TCL}$ then each structure $\mathcal{S}_n$ is a model of ϕ according to the definition of $\mathcal{S}_n$.

- If $\phi = \psi_{n_0}$ then a structure $\mathcal{S}$ is a model of ϕ iff there is an R^+-edge in $\mathcal{S}$ which is not justified by any R-path of length $l \leq n_0$. Obviously, $\mathcal{S}_{n_0+1}$ has this property, since per definition it encodes the graph G_{n_0+1}, in other words $I^{\mathcal{S}_{n_0+1}}(R^+) = R^+_{n_0+1} \supseteq R^{n_0+1}_{n_0+1} = \{(0, n_0+1)\}$. The R-path starting in 0 and leading to n_0+1 is the *only* R-path in the graph connecting 0 and n_0+1 and has length n_0+1. Thus, the minimal length of any R-path

in G_{n_0+1} connecting 0 and n_0+1 is $n_0+1 > n_0$. That means, the edge $(0, n_0+1)$ in G_{n_0+1} is not justified by any R-path of length $l \leq n_0$. Hence, $\mathcal{S}_{n_0+1}$ is a model of ϕ.

- If $\phi = \psi_n, n \in \mathbb{N}, \leq n_0$: Since $\psi_n \Rightarrow \psi_i$ for all $1 \leq i \leq n$, we can use the same argument as in the previous case $\phi = \psi_{n_0}$. In particular, the structure $\mathcal{S}_{n_0+1}$ is a model of ϕ.

To sum up, we have shown, that in all cases $\mathcal{S}_{n_0+1}$ is a model of $\phi \in \Phi_{n_0}$ and thus this structure is a model of Φ_{n_0}. Since Φ_{n_0} is satisfiable, so is Φ.

According to the compactness theorem, we can conclude that $\Phi^* = \{\phi^{TCL}\} \cup \overline{K^{TCL}}$ has to be satisfiable which (according to our construction) immediately leads to a contradiction: The models of $\overline{K^{TCL}}$ are precisely those Σ-structures $\mathcal{S}$ where there is an R^+-edge which is not justified by any R-path (of any length). That means that $I^{\mathcal{S}}(R^+)$ can not be the smallest transitive relation that contains $I^{\mathcal{S}}(R)$ and thus $I^{\mathcal{S}}(R^+) \neq (I^{\mathcal{S}}(R))^+$. On the other hand, according to our initial assumption, the models of ϕ^{TCL} are precisely Σ-structures $\mathcal{S}$ where $I^{\mathcal{S}}(R^+) = (I^{\mathcal{S}}(R))^+$. Hence, we have that Φ^* is unsatisfiable which contradicts our conclusion from the compactness theorem.

Thus, the inital assumption has been wrong and we have completed the proof. □

4 Discussion

In section 2, we have shown that the transitive closure of some relation can be precisely defined in Datalog. Then, we have shown in section 3 that this operator can not be described in First-order Logic. Since Datalog programs are closed formulas in the Horn-fragment of FOL only, which is a *proper* subset of the language of FOL, this seems to be somehow strange in the first place and even a contradiction.

In fact it is not and according to our proof, we can exactly explain why there is no contradiction. Let us come back to the fundamental question, what causes the difference between Datalog and First-Order Logic in regard of the expressivity of transitive closure?

First, we could ask what our Datalog program P means in classical FOL. The answer is straightforward: Rule R_1 specifies that in any model $\mathcal{S}$ of P the interpretation of R^+ contains the relation represented by predicate R. This corresponds to clause (1) in definition 1.1. Additionally, Rule R_2 ensures that each such model $\mathcal{S}$ is also transitive. Now what is missing? As we have seen in the proof of theorem 3.2 and we can easily read off from definition 1.1, the only aspect of possible models $\mathcal{S}$ which is not garantueed by the speficication P is that the interpretation $I^{\mathcal{S}}(R^+)$ is the *smallest* such relation. In our proof, we exploited precisely this aspect in order to show that K^{TCL} is not definable in FOL.

But why does program P work in Datalog? As we have already discussed in section 2, the main reason for not having ,,unnecessary" (resp. unjustified)

edges in $I^{\mathcal{S}}(R^+)$ the particular semantics of P in Datalog, which is different from the standard Tarski-style semantics of FOL: The definition of the semantics of P as the *minimal model* M_P which corresponds to the so-called *closed-world assumption* (CWA) garantuess that the interpretation of predicate R^+ in M_P actually is the smallest transitive relation which contains the original relation R.

In this respect, one could say, that the built-in fixed-point operator that is used for the computation of the semantics of a program can be used naturally to express the transitive closure.

In fact, it is well-known that certain mathematical properties resp. objects can not be precisely captured in a First-order Logic. A few examples are:

- The universe of the corresponding models is finite
- The universe of the corresponding models has an even number of elements
- Semantics of equality
- The property of graphs (resp. relations) of being connected
- Natural numbers
- More generally: *(General) inductive definitions*

Several logics that extend FOL in certain ways have been proposed in order to deal with the lack of expressivity of FOL. In particular the last mentioned example gave rise to a logic called *ID-Logic* [4], a logic which extends FOL with the possibility of inductive definitions. In fact, this logic seems to be naturally related to Logic Programming.

With respect to the last mentioned item in the list it might be interesting to point out, that usually in inductive definitions of properties, one states which elements have[1] the property to be defined and *implicitly* define all other elements not captured by the stated rules as not having the property under consideration. This is done by using a phrase like *,,M is the smallest set with ...”*.

There are several papers and articles which investigate the expressive power of Datalog [9], the relation between Datalog and First-order Logic [1], inductive definitions in logics [5, 4, 13, 12] and transitive closure and its role in inductive definitions [2].

[1] That means that we only consider one side of the story, namely the positive information resp. elements that have some property (in constrast to the description of elements that do not have that property).

4.1 How can we remedy this lack of expressiveness in First-order Logic?

In this section we show that some natural extensions of FOL provide the means for the description of the transitive closure property within the language itself.

4.1.1 Second-order Logics $\mathcal{L}_{II}$

Second-order logics [8] extend the syntax and semantics of First-order logics by means for quantification over *sets* of elements in the universe. In particular, there are additional variables $X, Y, Z, \ldots$ of a new category which allow to denote sets of elements.

By using the new description means, it becomes very easy to capture the minimality requirement on the interpretation of the predicate R^+ and hence to define the transitive closure in a Second-order Logic $\mathcal{L}_{II}$:

$$\begin{aligned}\phi^{TCL} \equiv\ & \forall x, y : R(x, y) \rightarrow R^+(x, y) \wedge \\ & \forall x, y, z : R^+(x, y) \wedge R^+(y, z) \rightarrow R^+(x, z) \wedge \\ & \forall X : [(\ \forall x, y : R(x, y) \rightarrow X(x, y) \wedge \\ & \qquad \forall x, y, z : X(x, y) \wedge X(y, z) \rightarrow X(x, z)) \\ & \qquad \rightarrow \forall x, y : R^+(x, y) \rightarrow X(x, y)]\end{aligned}$$

4.1.2 Infinitary logics like $\mathcal{L}_{\omega_1\omega}$

A simple approach for describing the semantics of the transitive closure of some relation in FOL would be as follows: As we have already seen, the properties (1) and (2) in definition 1.1 are easily definable in FOL. In order to capture minimality of the interpretation of R^+ (that means that we have no unjustified edges in the interpretation of this predicate wrt. to the interpretation of R), we use the following formula:

$$\begin{aligned}&\psi^{min} \equiv \\ &\quad \exists e, e' : (R^+(e, e') \wedge \\ &\quad \neg(\ (\exists e_0, e_1 : e_0 \doteq e \wedge e_1 \doteq e' \wedge R(e_0, e_1)) \vee \\ &\qquad (\exists e_0, e_1, e_2 : e_0 \doteq e \wedge e_2 \doteq e' \wedge R(e_0, e_1) \wedge R(e_1, e_2)) \vee \\ &\qquad (\exists e_0, e_1, e_2, e_3 : e_0 \doteq e \wedge e_3 \doteq e' \wedge R(e_0, e_1) \wedge R(e_1, e_2) \wedge R(e_2, e_3)) \vee \\ &\qquad \ldots \vee \\ &\qquad \ldots)\end{aligned}$$

Unfortunately, the resulting expression is an *infinite* one and thus no valid formula in a FOL!

The infinitary logic $\mathcal{L}_{\omega_1\omega}$ [10, 8, 3] extends FOL by allowing *infinite* disjunctions and conjunctions in formulas. Thus, the following $\mathcal{L}_{\omega_1\omega}$ formula ϕ_ω^{TCL} has precisely the class K^{TCL} as its models:

$\phi_{\omega}^{TCL} \equiv$
$\forall x, y : R(x, y) \rightarrow R^{+}(x, y) \wedge$
$\forall x, y, z : R^{+}(x, y) \wedge R^{+}(y, z) \rightarrow R^{+}(x, z) \wedge$
$\neg[\exists e, e' : (R^{+}(e, e') \wedge$
$\neg(\bigvee_{n \in \mathbb{N}} \exists e_0, \ldots, e_n : e_0 \doteq e \wedge e_n \doteq e' \wedge \bigwedge_{1 \leq i \leq n} R(e_{i-1}, e_i)))]$

4.1.3 Fixed-point logics like LFP

As we have seen in theorem 3.2 there are properties which can not be completely characterized in FOL. One class of such properties is properties which are defined by means of least fixpoints.

For instance the transitive closure R^{+} of some relation R can be defined as the least fixpoint of some relation R':

$$R^{+} = LFP_{R'}(R' = (R' \circ R) \cup R)$$

Since this is a restriction of FOLs with respect to expressivity (resp. as a knowledge representation framework) it's natural to look at extensions of FOL which provide syntactic means for the fixpoint operators. One such logic is LFP which is formed by closing FOL under the rule:

If ϕ is a formula, *positive* in the relational variable R, then so is $[\mathbf{lfp}_{R, \overrightarrow{x}} \phi](\overrightarrow{t})$ where $\overrightarrow{t}$ is a tuple of terms (in particular variables) and $\overrightarrow{x}$ is a tuple of (free) variables in ϕ.

The formula is read as: The tuple of elements denoted by the tuple $\overrightarrow{t}$ (wrt. to the considered structure $\mathcal{S}$) is in the least fixpoint of the operator $I^{\mathcal{S}}(R)$ to the set of tuples (of elements from $U^{\mathcal{S}}$) that satisfy $\phi(R, \overrightarrow{x})$.

Thus, the following LFP formula ϕ_{FLP}^{TCL} has precisely the class K^{TCL} as its models:

$\phi_{LFP}^{TCL} \equiv$
$\forall e, e' : (R^{+}(e, e') \leftrightarrow \ [\mathbf{lfp}_{T, xy}(R(x, y) \vee (\exists z : T(x, z) \wedge R(z, y)))](e, e'))$

References

[1] Miklos Ajtai and Yuri Gurevich. Datalog vs. first-order logic. In *In 30th Annual Symposium on Foundations of Computer Science*, pages 142–147. IEEE, November 1998.

[2] A. Avron. Transitive closure and the mechanization of mathematics. In F. Kamareddine, editor, *Thirty Five Years of Automating Mathematics*, pages 149–171. Kluwer Academic Publishers, 2003.

[3] Anuj Dawar, Steven Lindell, and Scott Weinstein. Infinitary logic and inductive definability over finite structures. *Information and Computation*, 119(2):160–175, 1995.

[4] Marc Denecker. Extending classical logic with inductive definitions. *Lecture Notes in Computer Science*, 1998.

[5] Marc Denecker. The well-founded semantics is the principle of inductive definition. *Logics in Artificial Intelligence*, 1998.

[6] G.Kreisel and J.-L. Krivine. *Modelltheorie*. Springer, 1972.

[7] J. Flum H.-D. Ebbinghaus. *Finite Model Theory*. Perspectives in Mathematical Logic. Springer, 2nd rev. and enlarged ed. edition, 1999.

[8] W. Thomas H.-D. Ebbinghaus, J. Flum. *Mathematical Logic*. Undergraduate Texts in Mathematics. Springer, 2nd ed. 1994. corr. 2nd printing edition, 1996.

[9] Phokion G. Kolaitis and Moshe Y. Vardi. On the expressive power of datalog: tools and a case study. In *Proceedings of the ninth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*, pages 61–71. ACM Press, 1990.

[10] H. Jerome Kreisler. *Model Theory for Infinitary Logic*. North-Holland Publishing Company – Amsterdam London, 1971.

[11] P. Lindström. On extensions of elementary logic. *Theoria*, 35:1–11, 1969.

[12] Maurice Bruynooghe Marc Denecker and V. Wiktor Marek. Logic programming revisited: Logic programs as inductive definitions. *ACM Trans. Comput. Log.*, 2001.

[13] L. C. Paulson and A. W. Smith. Logic programming, functional programming, and inductive definitions. 1989.

[14] A. Tarski. Contributions to the theory of models, i. *Indagationes Mathematicae*, 16:572–581, 1954.