INSTITUTE FOR LOGIC,
LANGUAGE AND COMPUTATION

EXPLORING THE CORE LOGIC OF FUNCTIONAL DEPENDENCE

Johan van Benthem, with Alexandru Baltag

ILLC, University of Amsterdam

TRENDS IN LOGIC19, Moscow, 2 Oct 2019

Dependence is Ubiquitous

Dependence of *x* on *y*, or of *x* on a set *Y*:

Correlated values (probability), Construction (vector spaces),

Dependent occurrence or behavior (causality, games).

Not necessarily one single notion!

Diverse senses of dependence in logic: $\forall x\ \exists y\ Rxy$, $\{\varphi, \varphi \wedge \psi\}$

Ontic coupled behavior in the physical world

Epistemic information about *x* gives information about *y*

We will explore one basic sense that applies to both.

A Dependence Table

Data base: tuples of values for attributes

Simple case: assignment of objects to variables

Global dependence

x	*y*	*z*	$x \to y$: *y depends on x*
0	**1**	**0**	$x \to y$, $y \to z$ (so: $x \to z$)
1	**1**	**0**	*not* $y \to x$ (so: *not* $z \to x$)
2	**0**	**0**	*not* $z \to y$

Local dependence *x* depends on *y* ***at*** (***2, 0, 0***)

but not at (***1, 1, 0***)

INSTITUTE FOR LOGIC, LANGUAGE AND COMPUTATION

Dependence Models

,

$$\mathcal{D} = (V, O, S, P)$$

V variables (in fact, any abstract objects)

O objects (possible values of variables)

S family of functions from *V* to *O*

need not be full O^V: **gaps = dependence**

P predicates of objects (to interpret predicate letters)

$D_X y$ for all *s*, *t* in *S*: if $s =_X t$, then $s =_y t$

with $s =_x t$: $s(x) = t(x)$, $s =_X t$: $\forall x \in X$: $s =_x t$

also lifted to sets $D_X Y$: for all $y \in Y$: $D_x y$

Variety of Interpretations

’ The preceding numerical example may be misleading.

Data bases can also assign non-numerical values.

Other interpretations of dependence models

Information: dependence as knowledge.

Interrogative: dependences between questions.

Processes: global states of dynamical system.

Games: objects are actions, dependent behavior.

Background: CRS-style First-Order Logic

Dependence models : ‘generalized assignment models’

$\boldsymbol{M} = (D, \mathcal{V}, I)$ with $\mathcal{V}$ set of ‘available’ assignments

$\boldsymbol{M}, s \models \exists x.\varphi$ iff there exists t in $\mathcal{V}$ with $s =^x t$ and $\boldsymbol{M}, t \models \varphi$

$s =^x t$: $s(y) = t(y)$ for all variables y distinct from x

Motivation Decouple (a) giving compositional semantics

for the first-order language, from the high cost:

(b) undecidable complexity of first-order valid laws.

Separate logical base from special mathematical structure.

Institute for Logic, Language and Computation

More Generalized First-Order Logic

Fact The first-order validities on generalized assignment models are recursively axiomatizable and decidable.

Drops independence principles such as $\exists x.\exists y.\varphi \rightarrow \exists y.\exists x.\varphi$: these impose existential confluence properties on the set $\mathcal{V}$.

Supports richer languages with tuple quantifiers $\exists \mathbf{x}.\varphi$.

H. Andréka, J. van Benthem & I. Németi, 1998, 'Modal Languages and Bounded Fragments of First-Order Logic', JPL.

The Basic Properties of Dependence

Fact *Dependence satisfies:*

$D_X x$ *for all* $x \in X$ — *Reflexivity*

$D_X z$ and $X \subseteq Y$ implies $D_y z$ — *Monotonicity*

$D_X Y$ and $D_Y Z$ implies $D_X Z$ — *Transitivity*

Thm Each reflexive monotonic transitive relation D is isomorphic to the dependence relation of some dependence model $\mathcal{D}$.

Connection to well-known Armstrong Axioms.

Representation, Sketch

For each set $X \subseteq \text{domain}(D)$, define two assignments s, t:

(i) For y with $D_X y$, $s(y) = (X, y)$, for y with $\neg D_X y$, $s(y) = (X, y, 1)$,

(ii) For y with $D_X y$, $t(y) = (X, y)$, for y with $\neg D_X y$, $t(y) = (X, y, 2)$.

Each true statement $D_U v$ is true in this dependence model $\mathcal{D}_X$.

Each false statement $D_X y$ is false in $\mathcal{D}_X$.

The disjoint union of all $\mathcal{D}_X$ has $D_U v$ true iff $D_U v$ is true in each separate $\mathcal{D}_X$. Hence, its dependence relation equals D.

Fact Similar results exist for local dependence.

Excursion: Consequence

How can this be? Reflexivity, Transitivity and Monotonicity are the characteristic properties of classical consequence.

Dependence is like consequence between **questions**: joint answer to premises implies answer to consequence.

Fact With two objects *0, 1*, dependence models cannot represent a strict linear order *x D y D z*, but consequence can.

To be determined Dependence logic for finite sizes of *O*.

Further Dependence and *non-classical* consequence relations.

INSTITUTE FOR LOGIC,
LANGUAGE AND COMPUTATION

From Implicit to Explicit Design

CRS reinterprets the language of first-order logic and packs information about (in-)dependence in failure of classical laws.

Alternative: introduce explicit vocabulary for dependence on the same models and keep the base logic classical.

We present the resulting system **LFD** in tandem: as a first-order logic, but equally well as a modal logic.

J. van Benthem, 2018, 'Implicit and Explicit Stances in Logic', JPL.

Language and Semantics LFD

Syntax φ ::= $Q\mathbf{x} \mid D_X y \mid \neg\varphi \mid \varphi \wedge \varphi \mid D_X\varphi$

existential dual quantifier/modality: $<D>_X\varphi$

Models $\mathcal{M} = (\mathcal{D}, Val)$, *Val* maps predicates into *P*

Truth definition

$\mathcal{M}, s \models Q\mathbf{x}$ iff $Val(Q)(s(\mathbf{x}))$, $\mathcal{M}, s \models D_X y$ iff $D^{\mathcal{D}}{}_X y$ at s

$\mathcal{M}, s \models D_X\varphi$ iff *for all t with* $s =_X t$, $\mathcal{M}, t \models \varphi$

So, our basic notion is **local** at some assignment.

Expressive Power

Defined notions

• Changing x implies changing y: $D_y x$

• Global senses included: $D_\emptyset\varphi$ is the universal modality $U\varphi$

Global dependence defined from local notion: $UD_X y$

• Dependence as value restriction (e.g., in databases):

if x lies within some range, then so does y: $U(Q_1 x \to Q_2 y)$

What is an optimal corresponding notion of **bisimulation**?

Options: dependence models have several moving parts.

Institute for Logic, Language and Computation

Fixing Variables and Locality

The variables that matter to a formula:

$$fix(P\mathbf{x}) = \{x_1, \cdots, x_k\}, \;\; fix(D_X y) = X$$

$$fix(\neg\varphi) = fix(\varphi), \; fix(\varphi \wedge \psi) = fix(\varphi) \cup fix(\psi)$$

$$fix(D_X\varphi) = X$$

Fact If $fix(\varphi) \subseteq X$, and $s =_X t$, then $\mathcal{M}, s \models \varphi$ iff $\mathcal{M}, t \models \varphi$

Induction on formulas, using properties of equivalence relations.

E.g., $D_x Py$ depends on current value of x, not on that of y.

INSTITUTE FOR LOGIC,
LANGUAGE AND COMPUTATION

Dependence and Explicit Definability

Another general intuition of dependence is ‘construction’:

y depends on objects *X* if *y* can be **defined** from the *X*

With a repertoire of functions, and complex terms *t*:

for all s: $s(y) = t(s(\mathbf{x}))$

Fact Functional definability implies semantic dependence.

Conversely,

Fact Semantic dependencies induce an algebra

of partial functions that witness them.

Institute for Logic, Language and Computation

Translation Into Standard First-Order Logic

Thm There is a translation ***tr*** from the language of LFD into first-order logic making the following equivalent for modal formulas φ:

(a) φ is satisfiable in a dependence model,

(b) ***tr***(φ) is satisfiable in a *standard* first-order model.

Trick as for translating CRS into guarded FOL: work with finitely many variables ***x***, and code that a tuple of values for ***x*** forms an available assignment with a new dedicated predicate *U****x***.

Corollary All logics that we will consider are RE (axiomatizable).

A Modal Perspective: Relational Models

Abstraction step:

objects not really needed for our language.

General relational models $\boldsymbol{M} = (S, \sim_X, V)$

$\sim_X$ equivalence relations,

if $\boldsymbol{x} \subseteq Y$, and $s \sim_Y t$, then $s \models Q\boldsymbol{x}$ iff $t \models Q\boldsymbol{x}$

Standard relational models

Also require that $\sim_X = \{\sim_x \mid x \in X\}$

Institute for Logic, Language and Computation

Some Model Theory

Now many notions from modal logic apply, such as

Fact Finite Model Property w.r.t. general relational models.

Thm Each standard relational model is *isomorphic* with a dependence model, and vice versa.

Prf Create objects as equivalence classes in Frege's style.

Thm Each general relational model is a *p-morphic image* of a standard relational model.

Prf Non-trivial combinatorial tree unraveling construction.

Analogies with Epistemic Logic

From dependence models to epistemic S5 models

Worlds ~ assignments, variables ~ agents, accessibility $\sim_x$ is $=_x$, valuation for atomic $P\mathbf{x}$, $D_X y$: y knows what the X-group knows.

From epistemic models to dependence models

Assignments induced by worlds $ass_w(x) = \{v \mid w \sim_x v\}$

Variables can stand for objects, agents, truth values of formulas.

Language analogies: $D_X\varphi$ is distributed group knowledge.

More: What is common knowledge as a dependence modality?

Tandem: First-Order and Modal View of LFD

In the perspective presented here,

modal and first-order are two sides of the same coin.

Therefore, notions and techniques apply either way.

Example: a route toward completeness and decidability

showing how a typical modal technique: **filtration**,

also applies to first-order logic.

But also, our **translation** yields compactness and strong completeness.

A First-Order Perspective on Filtration

Finite set of formulas F. Add all atoms $D_X y$ where the variables in X, y occur in F. Close under single negations. Result: finite set $\boldsymbol{F}$.

Given modal dependence model $\mathcal{M} = (\mathcal{D}, Val)$ and assignment s,

$$\boldsymbol{F}\text{-}type(\mathcal{M}, s) = \{\varphi \in \boldsymbol{F} \mid \mathcal{M}, s \models \varphi\}$$

The induced type model $type(\mathcal{M})$ consists of all such types,

and it is a finite family of finite sets.

This generates an interesting object-free 'quasi-model'.

Type Structures

Consider an induced type model $type(\mathcal{M})$.

Fact Types Σ satisfy the following for all ***F***-formulas :

(a) $\neg\varphi \in \Sigma$ iff not $\varphi \in \Sigma$, (b) $\varphi \wedge \psi \in \Sigma$ iff $\varphi \in \Sigma$ and $\psi \in \Sigma$

(c) if $D_X\varphi \in \Sigma$, then $\varphi \in \Sigma$, (d) if $\langle D\rangle_X\varphi \in \Sigma$, then there

exists a type Δ with $\varphi \in \Delta$ and $\Sigma \sim_X \Delta$, i.e.:

Σ, Δ agree on all formulas φ with $fix(\varphi) \subseteq \{y \mid D_X y \in \Sigma\}$

[in fact, the latter variables are the same in Σ and Δ]

These syntactic conditions define arbitrary ***F*-type structures**.

Representation, and Proof Sketch

Thm Each ***F***-type structure is induced by a dependence model.

Path Finite sequence π of types plus marked transitions $\sim_X$

Ass $ass_\pi(y)$ is the pair (π, y) if (a) $lth(\pi) = 1$, (b) $\pi = (\pi', \sim_X, \Sigma)$ where y does not depend on X according to Σ, else, (c) [still with $\pi = (\pi', \sim_X, \Sigma)$], $ass_\pi(y) = ass_{\pi'}(y)$

Key In the resulting dependence model, for all ***F***-formulas φ:

$ass_\pi \models \varphi$ iff $\varphi \in last\text{-}type(\pi)$

Induction on φ, variable chasing through forks in tree.

Decidability of LFD

Thm LFD is decidable.

Prf By the preceding results, being satisfiable is equivalent to occurring in some set in a finite type structure.

And it is clearly decidable whether a given modal formula occurs in some set in a finite type structure.

Open problem Does LFD have the Finite Model Property?

Open problem What is the computational complexity of LFD?

What about Standard First-Order Logic?

Fact FOL is undecidable.

But it is clearly decidable whether a given
first-order formula has a finite type structure.

Also, each type structure is representable
as a generalized first-order model.

Conclusion It is undecidable if a given type structure
is representable as a standard (full) first-order model.

Modal Deduction, Axiom System for LFD

The logic LFD consist of

(a) The principles of modal S5 for each separate $D_X\varphi$

(b) Monotonicity $D_X\varphi \rightarrow D_{X\cup Y}\varphi$

(c) Reflexivity, Transitivity, Monotonicity for atoms $D_X y$

(d) Transfer axiom $(D_X Y \wedge D_Y\varphi) \rightarrow D_X\varphi$

(e) Invariance $(\neg)Q\mathbf{x} \rightarrow D_{\mathbf{x}}(\neg)Q\mathbf{x}$, $(\neg)D_X y \rightarrow D_X(\neg)D_X y$

Fact The proof calculus for LFD is sound.

INSTITUTE FOR LOGIC, LANGUAGE AND COMPUTATION

Validity and Formal Derivation

Some practical dependence reasoning:

- Valid and derivable: $D_X D_Y \varphi \to D_{X \cup Y} \varphi$

  Invalid: $D_{X \cup Y} \varphi \to D_X D_Y \varphi$

- Fix Lemma derivable: $\varphi \to D_X \varphi$, if fix(φ) $\subseteq X$

- Invalid: for X, Y with $X \cap Y = \emptyset$,

  $(<D>_X \varphi \wedge <D>_Y \psi) \to <D>_{X \cup Y} (\varphi \wedge \psi)$

  But valid and derivable variants exist

Completeness

Thm The axiomatic proof calculus for LFD is complete.

Prf Consider any consistent formula with finite filtration set ***F***,
take all maximally consistent sets of formulas in ***F***,
and show that this family satisfies all conditions for a
type structure, especially the witness clause (d) for $<D>_X\varphi$.

All items in the proof system show their rationale in this argument.

Institute for Logic, Language and Computation

Sequent Calculus

- $$\frac{}{\Rightarrow D_X Y}$$ when $Y \subseteq X$
- $$\frac{\Gamma \Rightarrow \Delta, D_X Y \qquad \Gamma \Rightarrow \Delta, D_Y Z}{\Gamma \Rightarrow \Delta, D_X Z}$$
- $$\frac{\Gamma \Rightarrow \Delta, D_X Y \qquad \Gamma \Rightarrow \Delta, D_X Z}{\Gamma \Rightarrow \Delta, D_X Y \cup Z}$$
- $$\frac{\varphi, \Gamma \Rightarrow \Delta}{D_X \varphi, \Gamma \Rightarrow \Delta}$$
- $$\frac{\Gamma \Rightarrow \Delta, \varphi}{D_X Y, \Gamma \Rightarrow \Delta, D_X \varphi}$$ when $fix(\Gamma \cup \Delta) \subseteq Y$

Proof Theory

Thm The sequent calculus for LFD has cut elimination up to atomic dependence axioms.

Details in the full paper

Cor The sequent calculus and the axiomatic system for LFD have the same deductive power.

This also provides an alternative proof of decidability.

Open problem Can we also use it to prove *Interpolation*?

Correspondence for Additional Axioms

Invalid axioms can characterize special classes of models.

Analyzable via modal frame correspondence.

Fact $D_X D_Y \varphi \rightarrow D_Y D_X \varphi$ characterizes confluent models.

Cor The logic of commuting $D_X D_Y$ is undecidable.

Dependence atoms can also satisfy special laws:

$D_{X\cup\{y\}}\, z \rightarrow (D_X\, z \vee D_{X\cup\{z\}}\, y)$ Steinitz Exchange Principle

requirement on invertability of functional dependencies.

Language Extensions

Simple additions keep the logic decidable:

- CRS quantifiers, dual in a sense to LFD modalities.

Result: extended first-order logic over dependence models.

- Function terms: $x \mid f\,\boldsymbol{x}$ (still poor: e.g., $D_x fy$)

Open problem

Natural move: add explicit identities between terms.

Is LFD with identities between terms decidable?

Modal Logic of Independence

Independence is not the negation of dependence $\neg D_X y$.

Natural sense of independence of *y* from *X*:

Fixing the values of *X* leaves *y* free to take on any value it can take in the model ('knowing *X* implies no knowledge about *y*').

This can be formalized as an **independence modality** $\boldsymbol{I_X y}$.

Thm The modal logic of *I* is undecidable.

Prf Use statements $I_{\{x,\ y\}}z$ to force the range of *x*,*y*, *z* to form a Cartesian product, embed the three-variable fragment of FOL.

Dynamic Extensions

What changes in dependence models make intuitive sense?

Update changes models $\mathcal{M}$: **dependencies can change**.

Fact LFD with announcement of the actual value of *x* is decidable.

Prf The corresponding modal operator $[!x]\varphi$ satisfies obvious reduction axioms to the base language of LFD.

Open problem LFD + announcement $!\varphi$ of true facts decidable?

Reduction axioms need *conditional dependence modality*: $D_X^{\psi}\varphi$ plus atoms $D_X^{\psi}y$. RE by our translation, but is it decidable?

Universes for Model Change

Other model transformations in other settings. E.g.,
choice function for a variable ~ *strategy* in game.

Update takes place in families of dependence models:
dependence universes

Modalities for going to submodels, extensions, other links.

Distinguish accidental from essential dependencies:

$D_X y$ vs. $[\uparrow] D_X y$ ($D_X y \rightarrow [\downarrow] D_X y$ is valid)

Open problem What is the logic of LFD plus $[\uparrow]$, $[\downarrow]$?

Topological Version

Topological spaces, open sets as measurements,

approximating *y*-values by suitably restricting ***x***-values,

dependence ~ **continuous function**.

Semantics/model theory of LFD lifts to this setting:

Dependence models now have a topology ***O*** on values.

$\boldsymbol{M}, s \models D_X\varphi$ iff $\exists U \in O: s(X) \in U$ & $\forall t$ with $t(x) \in U$: $\boldsymbol{M}, t \models \varphi$

$\boldsymbol{M}, s \models D_X y$ iff $\forall U \in O$ s.t. $s(y) \in U$ $\exists V: \forall t$ with $t(X) \in V$: $t(y) \in U$

Topological Dependence Logic LCD

Complete axiom system like that for LFD,

but now on S4 basis for $D_X\varphi$.

Also, the further principles get different readings:

e.g., key Transfer Axiom expresses form of continuity:

$$(D_X Y \wedge D_Y\varphi) \rightarrow D_X\varphi$$

More radical approach Drop point values for variables.

dependence in point-free topology

Dependence in Vector Spaces

Dependence as constructibility (see earlier function definitions).

A different abstract analysis: **Matroid Theory**

Abstract mathematical representation of independent subsets.

Special role for dimension axiom (cf. Steinitz).

Fact (Gonzalez 2019) Each matroid can be represented

in a dependence model (matroid objects ~ variables).

Open problem

What modal language best matches matroid structure?

Dependence over Time

Many dependences take time. Tit-for-Tat in repeated games, or Copy-Cat in game semantics for linear logic: I play now what you played in the previous round.

Dynamical system with states (variable assignments) over time:

$$s_{t+1}(x) = F(s_t(y))$$

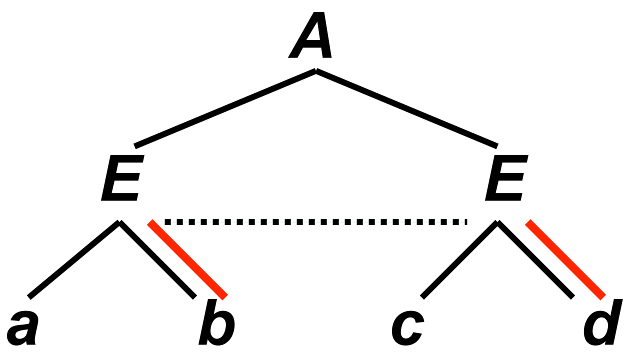
Suggests dynamical system over static dependence model, where the same assignment can return at different stages.

Open problem Design optimal temporal dependence logic.

Games, Choice and (In-)Dependence

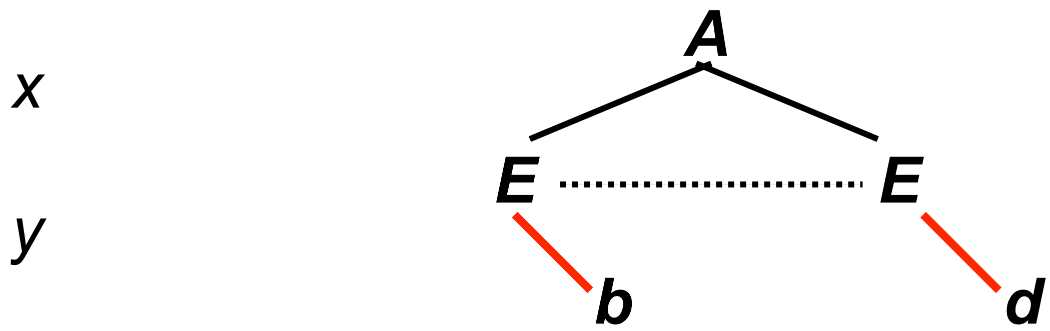
Games, dependent choices vs. dependent values,
imperfect information and independence (?),
uniform **strategies**: only depend on observed moves.

Aside: 'IF logic' is somewhat of a misnomer.



variables *x, y* for turn levels, object values are moves

Choice as Defining Minimal Model Change



$D_x y$ did not hold in the original game, but now it does

choice/strategy: minimal deletion of histories in game tree to make some specified dependence statements true, new notion of 'minimally changed submodel'.

Dynamic universe: logic of game play is **dynamic LFD**.

INSTITUTE FOR LOGIC, LANGUAGE AND COMPUTATION

Yet Further Notions of Dependence

situation theory in philosophy, Dretske's principles of information flow, channel theory Barwise & Seligman

causal graphs, imposing causal order on variables, reasoning about interventions by fixing values (connects to our representation theorem, and dynamic extensions)

And of course: **statistical correlation**

In progress Analyze as extensions of our framework.

Related Logical Work

- **CRS**, modal semantics for first-order logic
- van Lambalgen, **probabilistic independence**
- van den Berg, **plural semantics** with assignment sets
- Wang knowing-wh in **extended epistemic logic**
- **Inquisitive logic** for questions
- Väänänen, **dependence logic** (semantics over second-order models, quantifiers 'free variables from their dependencies')

Discussed in the full paper

Institute for Logic, Language and Computation

References

- H. Andréka, J. van Benthem & I. Németi, 'Modal Languages and Bounded Fragments of Predicate Logic', JPL 1998
- A. Baltag, 'Axiomatizing Modal Dependence Logics', AiML 2016
- A. Baltag & J. van Benthem, 'The Decidable Logic of Functional Dependency', ILLC UvA, 2019
- J. van Benthem, *Exploring Logical Dynamics*, 1996
- J. van Benthem & M-C Martinez, The Stories of Logic and Information, *Handbook of the Philosophy of Information*, 2009