

Local Dependence and Guarding

Johan van Benthem

ILLC University of Amsterdam, Stanford University, Tsinghua University

Balder ten Cate¹

ILLC University of Amsterdam

Raoul Koudijs

ILLC University of Amsterdam

Abstract

We study LFD, a base logic of functional dependence introduced by Baltag and van Benthem (2021) and its connections with the guarded fragment GF of first-order logic. Like other logics of dependence, the semantics of LFD uses teams: sets of permissible variable assignments. What sets LFD apart is its ability to express local dependence between variables and local dependence of statements on variables.

Known features of LFD include decidability, explicit axiomatization, finite model property, and a bisimulation characterization. Others, including the complexity of satisfiability, remained open so far. More generally, what has been lacking is a good understanding of what makes the LFD approach to dependence computationally well-behaved, and how it relates to other decidable logics. In particular, how do allowing variable dependencies and guarding quantifiers compare as logical devices?

We provide a new compositional translation from GF into LFD, and conversely, we translate LFD into GF in an ‘almost compositional’ manner. Using these two translations, we transfer known results about GF to LFD in a uniform manner, yielding, e.g., tight complexity bounds for LFD satisfiability, as well as Craig interpolation. Conversely, e.g., the finite model property of LFD transfers to GF. Thus, local dependence and guarding turn out to be intricately entangled notions.

Keywords: Logic, Dependence, Guarded Fragment

1 Introduction: from guarding to dependence

The Guarded Fragment GF of first-order logic, [2], is a well-known formalism for quantifying over tuples of objects that are locally *guarded* by predicates. Guarded existential quantification has the format $\exists \bar{y}(G(\bar{x}, \bar{y}) \wedge \varphi(\bar{x}, \bar{y}))$, where

¹ Supported by the European Union’s Horizon 2020 research and innovation programme under grant MSCA-101031081.

\bar{x} , \bar{y} are finite sets or tuples of variables, the atom $G(\bar{x}, \bar{y})$ is the guard, and $\varphi(\bar{x}, \bar{y})$ is a guarded formula having at most the free variables displayed. For further literature on GF and its extensions, see [7], [5].

Guarding quantifiers in truth conditions for existing semantics is a general device for lowering complexity of logical systems, [3]. But there are more such devices. In particular, [2] connects guarding with ‘generalized assignment models’ for first-order logic [to be called ‘teams’ henceforth in accordance with modern terminology] where not all maps from variables to objects need to be present. These ‘gaps’ in the full function space can be seen as modeling *dependence*, a major topic in the current logical literature, [1], [16]. With assignments missing, correlations arise, as changing the value of one variable x may only be feasible in the given space by also changing that of another variable y . In contrast, in standard models for first-order logic all variables take their values independently. The origins of this semantics lie in ‘cylindric relativized set algebra’ CRS in algebraic logic where standard set algebras are relativized to one fixed relation over the domain, [15]. Unlike first-order logic, CRS logic is decidable.

That guarding can encode dependence was proved in [2]. First-order formulas in CRS semantics translate compositionally into guarded formulas over standard first-order models by coding the available tuples of values for finite tuples of relevant variables \bar{v} as one new guard predicate $G(\bar{v})$. Conversely, [18] gave an effective reduction for the satisfiability problem for GF formulas φ to that for first-order formulas over team models, though the translation employed is not compositional.² Thus, dependence models can also encode guarding. The main topic of this paper is a further technical analysis of how far this analogy goes, now based on modern dependence logics rather than CRS.

CRS-style semantics treats dependence implicitly by giving up classical FO laws such as Commutativity $\exists x \exists y \varphi(x, y) \leftrightarrow \exists y \exists x \varphi(x, y)$ which express independence of the variables x and y . In a next step, [16] introduced explicit syntactic atoms $D_X y$ expressing functional dependence of variable y on the set of variables X . The first generation of dependence logics over this richer language was second-order and non-classical, but many simpler fragments have been studied. A systematic use of GF-style guarding to lower the complexity of dependence logics is found in [8]. However, the present paper is concerned with a simpler base system for dependence logic.

A new base logic LFD for a first-order language with explicit dependence atoms was recently proposed in [4] as a minimal way of reasoning about dependence on a classical base relying on a modal local semantics. LFD is a dualized CRS logic [in a sense to be defined below] plus dependence atoms, which is decidable, axiomatizable, and allows for natural language extensions, e.g., to function symbols and independence modalities. Many of these properties involve GF-style methods such as the use of type models and representation theorems. Thus, the question arises whether the connection between guarding

² Related results and further general analysis are found in [13].

and dependence extends to this new setting. But there are obstacles here. The first-order translation for LFD lands in the guarded fragment, except for the dependence atoms, cf. [9]. In this paper, we offer the following new results.

- We give a compositional translation from GF into LFD without dependence atoms, considerably improving the known SAT reduction.
- We give an effective SAT reduction from LFD into GF, answering a question left open in the literature.
- We show how these results allow for transfer of important properties between LFD and GF. For instance, the finite model property for GF [7], can be derived from that for LFD, established in [12]. Conversely, the decidability of LFD follows from that of GF. Moreover, using our translations, we solve the open problem of the computational complexity of LFD.
- Beyond single examples, we also discuss the general transfer of system properties made possible by our translations, and discuss more general consequences for relating decidable fragments of first-order logic.

All missing proofs can be found in the appendix.

2 Preliminaries

2.1 The Logic of Functional Dependence LFD

The Logic of Functional Dependencies (LFD) was introduced in [4], to which we refer for all definitions and results stated in this section.

Definition 2.1 (Syntax of LFD) Fix a finite set of variables V_{LFD} , and a relational signature (i.e., a set of relation symbols with associated arity) \mathbf{S} . The formulas of LFD are recursively generated by the following grammar.

$$\psi ::= P(v_1, \dots, v_n) \mid D_V u \mid \psi \wedge \psi \mid \neg \psi \mid \mathbb{E}_V \psi$$

with $v_1, \dots, v_n, u \in V_{LFD}$, $V \subseteq V_{LFD}$, and $P \in \mathbf{S}$ an n -ary relation symbol.

We read $D_V u$ as ‘ u locally depends (only) on V ’ or ‘ u is locally determined by V ’. For notational convenience, we write $D_V U$ for the conjunction $\bigwedge_{u \in U} D_V u$. We also write $D_v u$ as a shorthand for $D_{\{v\}} u$, and $\mathbb{E}_v \psi$ for $\mathbb{E}_{\{v\}} \psi$.

The semantics of LFD uses *dependence models*, also known as ‘generalized assignment models’. These are pairs $\mathbb{M} = (M, A)$ where M is a standard model over the relational signature \mathbf{S} , and $A \subseteq \text{dom}(M)^{V_{LFD}}$ is a collection of admissible variable assignments, also known as a ‘team’.

Definition 2.2 (Semantics of LFD)

Truth of a formula φ in a dependence model $\mathbb{M} = (M, A)$ under an assignment $s \in A$ is defined as follows, where $s =_V t$ means that $s(v) = t(v)$ for all $v \in V$:

$$\begin{array}{ll}
\mathbb{M}, s \models P(v_1, \dots, v_n) & \text{iff } s(v_1, \dots, v_n) \in P^M \\
\mathbb{M}, s \models \varphi \wedge \psi & \text{iff } \mathbb{M}, s \models \varphi \text{ and } \mathbb{M}, s \models \psi \\
\mathbb{M}, s \models \neg\varphi & \text{iff not } \mathbb{M}, s \models \varphi \\
\mathbb{M}, s \models \mathbb{E}_V\varphi & \text{iff for some } t \in A, s =_V t \text{ and } \mathbb{M}, t \models \varphi \\
\mathbb{M}, s \models D_V u & \text{iff for all } t \in A, s =_V t \text{ implies } s =_u t
\end{array}$$

LFD formulas come with a notion of *free variable*, being a variable whose value can affect the truth of the formula, in the semantics to follow.

Definition 2.3 (Free Variables)

The free variables of an LFD formula ψ are defined as follows:

$$\begin{array}{ll}
\text{free}(P(v_1 \dots v_n)) & = \{v_1, \dots, v_n\} \\
\text{free}(\neg\varphi) & = \text{free}(\varphi) \\
\text{free}(\varphi \wedge \psi) & = \text{free}(\varphi) \cup \text{free}(\psi) \\
\text{free}(D_V u) = \text{free}(\mathbb{E}_V\psi) & = V
\end{array}$$

It might seem odd that $\text{free}(\mathbb{E}_x\varphi) = \{x\}$ while $\text{free}(\exists x\varphi) = \text{free}(\varphi) \setminus \{x\}$. The difference, however, is a matter of perspective, since the two notions of free variable coincide on their interpretation as “a variable whose value can affect the truth of the formula”. With the preceding definitions, it is easy to show that LFD, just like FO, satisfies a property called *variable locality*: any two assignments s, t that agree on some set of variables X agree on the truth values of all formulas whose free variables are contained in X (plus all variables that locally depend on X at s, t).

The difference with FO free variables arises because the LFD modality is dual to that of FO, and displays variables whose values are kept fixed, rather than those whose values are allowed to vary. More generally, LFD shares a feature with CRS that makes it distinct from FO in its standard semantics. While in first-order formulas quantified variables are arbitrary placeholders, in LFD formulas variables have an individual character given their possibly different roles in the assignment space.³

Standard first-order models induce special ‘full’ dependence models (M, M^V) where *all* assignments are available. On such models there are no non-trivial dependencies (atoms $D_X y$ are always false unless $y \in X$) and the dependence quantifier \mathbb{E} collapses to the standard existential quantifier \exists , recapturing FO. Conversely, dependence models can also be encoded as standard models, supporting a faithful translation into an expanded version of FO.

Concretely, there is a translation $tr(\cdot)$ from LFD formulas to FO formulas (over an expanded signature), and a contravariant function T that bijectively maps models of $tr(\varphi)$ to dependence models satisfying φ . In a picture:

³ Accordingly, renaming bound variables in one model is not generally possible in LFD, though one can shift to new variables when allowing changing the current model [4].

$$\begin{array}{ccc}
\text{LFD-formula } \varphi & \xrightarrow{tr} & \text{first-order formula } tr(\varphi) \\
\text{dependence model } T(M) & \xleftarrow[\text{(bijective)}]{T} & \text{first-order model } M
\end{array}$$

Since we will see variants of this diagram of contravariant maps in later sections of this paper, we will spell things out in detail, as it will serve as an example for our later translation schemes.

Definition 2.4 (First-Order Translation)

Let $V_{LFD} = \{v_1, \dots, v_n\}$. The translation function $tr(\cdot)$ maps LFD formulas over V_{LFD} to FO formulas using the same variables v_1, \dots, v_n and auxiliary variables v'_1, \dots, v'_n , and using an additional n -ary relation symbol A .

$$\begin{aligned}
tr(P(u_1, \dots, u_n)) &:= P(u_1, \dots, u_n) \\
tr(\varphi \wedge \psi) &:= tr(\varphi) \wedge tr(\psi) \\
tr(\neg\varphi) &:= \neg tr(\varphi) \\
tr(D_V u) &:= \forall \bar{v}' (A(\bar{v}') \wedge \bigwedge_{v \in V} (v = v' \rightarrow u = u')) \\
tr(\mathbb{E}_V \psi) &:= \exists \bar{z} (A(\bar{v}) \wedge tr(\psi))
\end{aligned}$$

where $\bar{v} = v_1, \dots, v_n$, $\bar{v}' = v'_1, \dots, v'_n$, and \bar{z} enumerates $V_{LFD} \setminus V$.

In this translation A is a new predicate encoding the admissible assignments on the relevant variables.

Definition 2.5 (Model Transformation)

Let $V_{LFD} = \{v_1, \dots, v_n\}$. For a standard model M over the extended signature with the A relation, $T(M)$ is the dependence model obtained by extracting a team $A := \{s : V \rightarrow M \mid (s(v_1), \dots, s(v_n)) \in A^M\}$ on M and removing the relation A^M from the model.

Note that the map T is bijective. Its inverse T^{-1} maps a dependence model $\mathbb{M} = (M, A)$ to the standard model over the extended signature obtained by interpreting A as $\{(s(v_1), \dots, s(v_n)) \mid s \in A\}$.

Theorem 2.6 *For every LFD formula ψ , standard model M , assignment $s : \text{free}(\psi) \rightarrow M$ such that $M, s \models A(v_1, \dots, v_n)$, we have the equivalence*

$$T(M), s \models \psi \quad \text{iff} \quad M, s \models tr(\psi) \wedge A(v_1, \dots, v_n)^4$$

Since $T(\cdot)$ is bijective, Theorem 2.6 implies, in particular, that $tr(\varphi) \wedge A(v_1, \dots, v_n)$ and φ are equi-satisfiable. As consequences of this translation, one can also derive an effective reduction of semantic consequence, Compactness, Löwenheim-Skolem, recursive enumerability, all of them inherited by LFD from FO. Other properties require additional work in reproving classical results, much as happens with the standard FO translation for modal logic [6].

⁴ The added conjunct $A(v_1, \dots, v_n)$ makes sure that s is an admissible assignment in M . We could also work this information into the compositional clauses of the translation.

Remark 2.7 (LFD and CRS) *Dependence models are the generalised semantic structures first introduced for the algebraic version CRS of FO logic, [15], with $M, s \models \exists x \varphi$ if some available assignment $t =^x s$ in M satisfies φ . Here the relation $=^x$ is dual to the $=_x$ used for LFD, requiring that s, t agree on the values for all variables except x . Thus CRS quantifiers, like the first-order ones, display the variables that are allowed to vary, rather than those kept fixed. A polyadic version $\exists \bar{x} \varphi$ is defined analogously, which, unlike in standard FO semantics, does not reduce to iterated single existential quantifiers. When the total set V of variables is finite, the existential quantifiers of CRS and the existential modality of LFD are intertranslatable, e.g., $\exists x \varphi = \mathbb{E}_{V \setminus \{x\}} \varphi$. Thus, the fragment of LFD without dependence atoms amounts to a version of CRS with ‘dual quantifiers’, more detailed discussion can be found in [4]. Thus, some results in what follows also specialize to new connections between GF and CRS.*

2.2 Dependence Bisimulations, and Type Models

We need to introduce two more technical notions related to LFD that will play an important role in our proofs in subsequent sections.

Given its similarities with modal logic, LFD comes with a notion of *dependence-bisimulations*, which leaves the truth values of formulas invariant [9,12]. For each set of variables V , dependence model \mathbb{M} , and assignment s , the *dependence-closure* $D_V^s = \{u \in V_{LFD} \mid \mathbb{M}, s \models D_V u\}$ is the set of variables locally determined by V at s . Clearly, $V \subseteq D_V^s$. We call a set V *dependence-closed* at assignment s if $D_V^s = V$. Also, for any two assignments, $V^{s,t} = \{v \in V_{LFD} \mid s =_v t\}$ denotes the set of variables on which s, t agree.

Definition 2.8 (Dependence Bisimulations) Let $(M, A), (M', A')$ be dependence models. A relation $Z \subseteq A \times A'$ is a *dependence bisimulation* if for every pair $(s, s') \in Z$, the following conditions are satisfied:⁵

- (atom)** $s(\bar{u}) \in P^M$ iff $s'(\bar{u}) \in P^{M'}$ for all tuples \bar{u} of LFD variables
- (forth)** for every $t \in A$, there is a $t' \in A'$ such that (i) $V^{s,t} \subseteq V^{s',t'}$,
(ii) $(t, t') \in Z$ and (iii) the set $V^{s,t}$ is dependence-closed at s'
- (back)** for every $t' \in A'$, there is a $t \in A$ such that (i) $V^{s',t'} \subseteq V^{s,t}$,
(ii) $(t, t') \in Z$ and (iii) the set $V^{s',t'}$ is dependence-closed at s

LFD formulas do not distinguish between dependence-bisimilar models. Indeed, it was shown in [9] that a first-order formula is invariant for dependence bisimulations iff it is equivalent to the $tr(\cdot)$ translation of an LFD formula.⁶

Dependence bisimulations allow us to show that LFD is ‘blind’ for precisely how assignments assign values to different variables. This is captured formally

⁵ Note that dependence bisimulations Z are always *total*, i.e., $dom(Z) = A$ and $cod(Z) = A'$.

⁶ The definition of LFD-bisimulations in [9] is stated differently: it does not have the dependence-closed condition (iii) in the forth and back-clauses but instead includes the requirement that $s \models D_X y$ iff $s' \models D_X y$ into the atomic condition. As it turns out, these definitions are equivalent, but the one in [9] also works for type models.

by the *distinguished model property of LFD*. A dependence model is *distinguished* if each variable only takes values in its own range different from the ranges of all other variables. Thus for every object in the domain there is a unique variable to which it can be assigned by an admissible assignment.

Proposition 2.9 *There is a transformation $(\cdot)^d$ taking any dependence model M into a dependence-bisimilar distinguished dependence model M^d .*

Proof. For each admissible assignment $s \in A$, let s^d be the assignment defined by $s^d(v) := (v, s(v))$. Now take the team $A_d := \{s^d \mid s \in A\}$ on the domain $\bigcup_{s \in A} s^d[V_{LFD}]$. We interpret relations on the new value domain by setting

$$R^{M_d} := \{(u_1, s(u_1)), \dots, (u_m, s(u_m)) \mid (s(u_1), \dots, s(u_m)) \in R^M\},$$

making it into a standard model M_d . Note that each $s^d : V_{LFD} \rightarrow M_d$ and hence (M_d, A_d) is a dependence model which is distinguished by construction. It is now straightforward to check that the relation $\{(s, s_d) \mid s \in A\}$ is a dependence bisimulation between (M, A) and (M_d, A_d) . \square

The distinguished model property has to do with the lack of an equality in the language. LFD extended with equality atoms $v_i = v_j$ is undecidable [9].

Finally, we review an abstract type model semantics, also known as ‘quasi-models’ [19,18], originally used to prove the decidability of LFD in analogy with type models for GF. One can think of type models as finite certificates for satisfiability, or as information reductions of ‘real’ models with objects.

Definition 2.10 (Types)

The *closure* of an LFD formula ψ , denoted $Cl(\psi)$, is the smallest set containing ψ that is closed under subformulas and single negations⁷ and that contains all atomic formulas of the form $D_V u$. A subset $\Delta \subseteq Cl(\psi)$ is a *type* (for ψ) if it satisfies the following three properties for the logical operations plus two further constraints on dependence atoms:

(\neg -Consistency) $\neg\chi \in \Delta$ iff not $\chi \in \Delta$

(\wedge -Consistency) $\chi \wedge \xi \in \Delta$ iff $\chi \in \Delta$ and $\xi \in \Delta$

(\mathbb{E} -Consistency) $\chi \in \Delta$ implies $\mathbb{E}_V \chi \in \Delta$ whenever $\mathbb{E}_V \chi \in Cl(\psi)$

(Projection) $D_V u \in \Delta$ for all $V \subseteq V_{LFD}$ and $u \in V$

(Transitivity) $D_V U, D_U W \in \Delta$ implies $D_V W \in \Delta$ for all $V, U, W \subseteq V_{LFD}$

For types Δ, Δ' and sets $V \subseteq V_{LFD}$, we will henceforth write $\Delta \sim_V \Delta'$ if $\{\psi \in \Delta \mid \text{free}(\psi) \subseteq V\} = \{\psi \in \Delta' \mid \text{free}(\psi) \subseteq V\}$.

Definition 2.11 (LFD Type Models)

Fix an LFD formula ψ . An LFD type model \mathfrak{M} is a set of types (for ψ) satisfying the following two conditions:

(witness) if $\mathbb{E}_V \varphi \in \Delta$ then $\exists \Delta' \in \mathfrak{M}$ with $\Delta \sim_{D_V} \Delta'$ and $\varphi \in \Delta'$

⁷ The single negation $\sim(\psi) = \psi'$ if $\psi = \neg\psi'$ for some ψ' , and $\sim(\psi) = \neg\psi$ otherwise.

(universal) \sim_{\emptyset} is the universal relation on \mathfrak{M}

with $D_V^{\Delta} = \{u \in V_{LFD} \mid D_V u \in \Delta\}$ the ‘dependence-closure’ of V w.r.t Δ .⁸

We say \mathfrak{M} is a type model for ψ if there is a type $\Delta \in \mathfrak{M}$ with $\psi \in \Delta$. As it turns out, LFD type models always encode the existence of a real dependence model, so they may serve as certificates for the satisfiability of an LFD formula.

Theorem 2.12 *There is an ‘unravelling’ operation R such that, for each LFD type model \mathfrak{M} , $R(\mathfrak{M})$ is a dependence model realizing precisely the types in \mathfrak{M} .*

In fact, there is a dependence bisimulation (in the sense of [9]) from $R(\mathfrak{M})$ onto \mathfrak{M} . Theorem 2.12 implies that LFD is also complete w.r.t. type models.⁹

2.3 The Guarded Fragment of First-Order Logic

By a “standard model” we mean an ordinary first-order structure, and for FO formulas φ , $\varphi(\bar{x})$ indicates that $\text{free}(\varphi) \subseteq \{\bar{x}\}$. We will be concerned with the Guarded Fragment GF [2]. This is the fragment of FO consisting of formulas in which all quantification is *guarded*, i.e., of the form $\exists \bar{y}(G(\bar{x}, \bar{y}) \wedge \psi(\bar{x}, \bar{y}))$, where $G(\bar{x}, \bar{y})$ is an atomic formula and $\text{free}(\psi) \subseteq \text{free}(G(\bar{x}, \bar{y})) = \{\bar{x}, \bar{y}\}$. We will write guarded quantifiers $\exists \bar{y}(G(\bar{x}, \bar{y}) \wedge \varphi)$ when it is clear from context that φ is guarded. We also use $\forall \bar{y}(G(\bar{x}, \bar{y}) \rightarrow \psi(\bar{x}, \bar{y}))$ as syntactic sugar for $\neg \exists \bar{y}(G(\bar{x}, \bar{y}) \wedge \neg \psi(\bar{x}, \bar{y}))$. For simplicity, we will be working with the *equality-free* guarded fragment. Our results extend to full GF with equality, cf. Section 6.

For later reference, observe that the translation $tr(\cdot)$ from LFD to FO that we discussed earlier lands in GF except for the dependence atoms: the formula $tr(D_X y)$ uses unguarded quantification.

For the properties of the Guarded Fragment used in this paper, we refer to the earlier-mentioned sources. In particular, the satisfiability problem for GF is decidable, while the language has a characterization as a fragment of FO in terms of invariance for *guarded bisimulations*, [2].

3 A compositional translation from GF into LFD

It was shown in [18] that satisfiability in the Guarded Fragment reduces effectively to satisfiability in the logic CRS. The latter, as we mentioned before, corresponds to the fragment of LFD without dependence atoms. The translation achieving this used conjunctions of (a) a compositional translation for GF formulas φ , (b) a ‘set-up component’ that records the requirements for a GF type model for φ , both stated as CRS formulas. In contrast, here, we will present a compositional translation from GF to LFD (in fact, to the fragment of the logic LFD without dependence atoms), allowing us to transfer properties such as Craig interpolation. Our surprisingly simple translation also neatly exhibits a conceptual difference between first-order variables and LFD variables.

Remark 3.1 (Two views of variables) *Given the duality between the LFD modality \mathbb{E}_X and the first-order quantifier $\exists X$, it is tempting to translate FO*

⁸ Note that if $\Delta \sim_V \Delta'$ then $D_V^{\Delta} = D_V^{\Delta'}$ because $\text{free}(D_V u) = V$.

⁹ This analysis can be rephrased modal-style in terms of ‘general relational models’ [4].

formulas to LFD by replacing $\exists x\varphi$ by $\mathbb{E}_Y\varphi$ with Y the set of all free variables of φ except x . But this does not work. Consider the unsatisfiable first-order sentence $\exists xP(x) \wedge \neg\exists yP(y)$. Its translation, following the recipe just described, is $\mathbb{E}_\emptyset P(x) \wedge \neg\mathbb{E}_\emptyset P(y)$. However, this LFD formula is satisfied in a dependence model $\mathbb{M} = (M, A)$ where M has domain $\{a, b\}$ with just one atomic fact $P(a)$, and every assignment in A maps x to a and y to b . The crux lies in the fact that, in first-order logic, variables are interchangeable placeholders, whereas in LFD, variables have individual behavior, at least within a dependence model.

To deal with this discrepancy, the following translation τ separates the sets of relevant variables V_{LFD} used for LFD and V_{GF} for GF, and then connects these explicitly via a mapping that can be seen as an abstract substitution.

Definition 3.2 Let φ be a FO formula and $\rho : \text{free}(\varphi) \rightarrow V_{LFD}$. The translation $\tau_\rho(\varphi)$ is defined as follows, with $\rho =_X \rho'$ for $\rho(x) = \rho'(x)$ for all $x \in X$:

$$\begin{aligned} \tau_\rho(P(x_1, \dots, x_n)) &:= P(\rho(x_1), \dots, \rho(x_n)) \\ \tau_\rho(\varphi \wedge \psi) &:= \tau_\rho(\varphi) \wedge \tau_\rho(\psi) \\ \tau_\rho(\neg\varphi) &:= \neg\tau_\rho(\varphi) \\ \tau_\rho(\exists \bar{y} \varphi(\bar{x}, \bar{y})) &:= \bigvee_{\substack{\rho' : \{\bar{x}, \bar{y}\} \rightarrow V_{LFD} \\ \rho =_{\bar{x}} \rho'}} \mathbb{E}_{\rho(\bar{x})} \tau_{\rho'}(\varphi) \end{aligned}$$

With this new compositional translation, and assuming that $V_{LFD} = \{v_1, v_2\}$, the above FO formula $\exists xP(x) \wedge \neg\exists yP(y)$ translates to the LFD-formula $(\mathbb{E}_\emptyset P(v_1) \vee \mathbb{E}_\emptyset P(v_2)) \wedge \neg(\mathbb{E}_\emptyset P(v_1) \vee \mathbb{E}_\emptyset P(v_2))$, which, indeed, is unsatisfiable. Even so, τ is not an effective satisfiability reduction from FO to LFD, since LFD is decidable. But as we will show, τ preserves satisfiability for formulas in the Guarded Fragment.

Our next step is to define a corresponding operation on models. The overall scheme will look like this:

$$\begin{array}{ccc} \text{GF-formula } \varphi & \xrightarrow{\tau} & \text{LFD formula } \tau_\rho(\varphi) \\ \text{standard model } G(\mathbb{M}) & \xleftarrow[\text{(surjective)}]{G} & \text{dependence model } \mathbb{M} \end{array}$$

The model transformation G , consists of throwing away 'unnamed' facts.

Definition 3.3 Given any dependence model $\mathbb{M} = (M, A)$, the standard model $G(\mathbb{M})$ has for its domain $\text{dom}(M)$ and for its predicate interpretations $R^{G(\mathbb{M})} := \{\bar{m} \in R^M \mid \{\bar{m}\} \subseteq s[V_{LFD}] \text{ for some } s \in A\}$, where $s[V_{LFD}]$ is the image of $s : V_{LFD} \rightarrow M$, i.e. $s[V_{LFD}] = \{s(v) \mid v \in V_{LFD}\}$.

The transformation G is surjective because each standard model M is the G image of its matching 'full' dependence model $F(M) = (M, M^{V_{LFD}})$. Note that, while $G(F(M)) = M$, in general $F(G(\mathbb{M}))$ can be very different from \mathbb{M} , since $F(G(\mathbb{M}))$ is always full. A further complication is that, while the

operation G is defined on all dependence models, the crucial equivalence theorem that connects τ with G , which we will present next, applies only for *distinguished* dependence models. We will address this issue separately.

Theorem 3.4 *For every distinguished dependence model $\mathbb{M} = (M, A)$, assignment $s \in A$ and GF formula φ with map $\rho : \text{free}(\varphi) \rightarrow V_{LFD}$ we have*

$$\mathbb{M}, s \models \tau_\rho(\varphi) \quad \text{iff} \quad G(\mathbb{M}), s \circ \rho \models \varphi$$

Proof. By induction on the complexity of the formula φ . The atomic case says that $(M, A), s \models P\rho(x_1)\dots\rho(x_n)$ holds iff $G(M, A), s \circ \rho \models Px_1\dots x_n$. This holds because, by our definition, the fact $P(s(\rho(x_1)), \dots, s(\rho(x_n)))$ does not get removed from (M, A) by the transformation G . The Boolean cases are routine, so the only important case to analyze is that of guarded quantification.

From left to right, suppose $(M, A), s \models \tau_\rho(\exists \bar{y}(G(\bar{x}, \bar{y}) \wedge \varphi))$ (where $\rho : \{\bar{x}\} \rightarrow V_{LFD}$). This means that there is some $\rho' : \{\bar{x}, \bar{y}\} \rightarrow V_{LFD}$ extending ρ such that $(M, A), s \models \mathbb{E}_{\rho[\bar{x}]}(\tau_{\rho'}(G(\bar{x}, \bar{y}) \wedge \varphi))$, since one of the disjuncts in our translation clause is true. By the LFD semantics, there is then some witnessing assignment $t \in A$ with $s =_{\rho[\bar{x}]} t$ and $t \models \tau_{\rho'}(G(\bar{x}, \bar{y}) \wedge \varphi)$. By the inductive hypothesis, it now follows that $G(M, A), t \circ \rho' \models G(\bar{x}, \bar{y}) \wedge \varphi$. But since $\rho[\bar{x}] = \rho'[\bar{x}]$ and $s =_{\rho[\bar{x}]} t$ also $s \circ \rho =_{\bar{x}} t \circ \rho'$. The GF semantics yields that $G(M, A), s \circ \rho \models \exists \bar{y}(G(\bar{x}, \bar{y}) \wedge \varphi)$ where the objects $(t \circ \rho')[\bar{y}]$ witness the bound variables \bar{y} .

From right to left, we need to appeal to the guard in the quantification. Suppose that $G(M, A), s \circ \rho \models \exists \bar{y}(G(\bar{x}, \bar{y}) \wedge \varphi)$. By the standard FO semantics, there are objects $\bar{m} = (m_1, \dots, m_n)$ s.t. $G(M, A), (s \circ \rho)[\bar{m}/\bar{y}] \models G(\bar{x}, \bar{y}) \wedge \varphi$. By distinguishedness, for each $m \in \bar{m}$ there is a *unique* LFD-variable v_m s.t. there is some admissible assignment $s \in A$ with $s(v_m) = m$. Now consider the true guard fact $G((s \circ \rho)(\bar{x}), \bar{m})$ (which is a fact of $G(M, A)$). Given the uniqueness of variable names in M , this fact must have been witnessed by some available assignment $t \in A$ using some atomic formula $G(\rho(\bar{x}), v_{\bar{m}})$, where $t \circ \rho(x) = s \circ \rho(x)$ for all $x \in \bar{x}$. It follows that $s =_{\rho(\bar{x})} t$. Now let $\rho' : \{\bar{x}, \bar{y}\} \rightarrow V_{LFD}$ extend ρ by mapping $y_i \mapsto v_{m_i}$ for each $1 \leq i \leq n$. It follows that (i) $s =_{\rho(\bar{x})} t$ in the model (M, A) , and (ii) $t \circ \rho' =_{\{\bar{x}, \bar{y}\}} s \circ \rho[\bar{m}/\bar{y}]$ as assignments in the model $G(M, A)$. In particular then, since the truth of a FO-formula under an assignment depends only on what the assignment maps its free variables to, $G(M, A), t \circ \rho' \models G(\bar{x}, \bar{y}) \wedge \varphi$. Applying the inductive hypothesis, we see that $M, t \models \tau_{\rho'}(G(\bar{x}, \bar{y}) \wedge \varphi)$ and hence $M, s \models \mathbb{E}_{\rho(\bar{x})}(\tau_{\rho'}(G(\bar{x}, \bar{y}) \wedge \varphi))$. This is one of the disjuncts in the translation $\tau_\rho(\exists \bar{y}(G(\bar{x}, \bar{y}) \wedge \varphi))$, and we are done. \square

As noted before, the map G from dependence models to standard models is surjective. However, not every standard model is the G -image of a *distinguished* dependence model. This is an issue, because Theorem 3.4 applies only to distinguished dependence models. Fortunately, we have the following:

Theorem 3.5 *Let the number of variables in V_{LFD} be at least as great as the maximum arity of relations in the signature. Then, for every standard model M and guarded assignment s , there is a distinguished dependence model \mathbb{M} and*

an admissible LFD assignment t in \mathbb{M} such that $(G(\mathbb{M}), t \circ \rho)$ is GF-bisimilar to (M, s) , for some map ρ .

Proof. Let $\mathbb{M} = (M', A)$ where M', A are as follows:

- $Dom(M') = Dom(M) \times V_{LFD}$, and
- $((m_1, u_1) \dots, (m_k, u_k)) \in R^{M'}$ iff $(m_1, \dots, m_k) \in R^M$ and for all $i, j \leq k$, if $m_i \neq m_j$ then $u_i \neq u_j$.
- $A \subseteq Dom(M')^{V_{LFD}}$ consist of all assignments that map each LFD-variable v to a pair of the form (m, v) for $m \in dom(M)$ some object in the old domain.

It is clear from the construction that \mathbb{M} is distinguished.

Since s is a guarded assignment, there exists a tuple $(m_1, \dots, m_k) \in R^M$ for some relation R , such that s maps every variable to an element of this tuple. We know that $|V_{LFD}| \geq k$. Let ρ be an arbitrary mapping from FO variables to LFD variables, such that $s(x) \neq s(y)$ iff $\rho(x) \neq \rho(y)$. Finally, let t be an LFD assignment that maps each $\rho(x)$ to the pair $(s(x), \rho(x))$. In this way, we have that, for each FO variable x , $(t \circ \rho)(x) = (s(x), \rho(x))$. Furthermore, by construction, t is an admissible assignment for \mathbb{M} .

It remains only to show that $(G(\mathbb{M}), t \circ \rho)$ is GF-bisimilar to (M, s) . The GF bisimulation in question consists of all partial functions $f : Dom(M') \rightarrow Dom(M)$ such that (i) $dom(f)$ is a guarded subset of $dom(M')$, and (ii) f is the natural projection on its domain, i.e., $f((m, v)) = m$. It can be easily verified that this satisfies all requirements of a GF-bisimulation. \square

We state one immediate consequence, further implications of our translations will be discussed in Section 5 below.

Corollary 3.6 *Let the number of variables in V_{LFD} be at least as great as the maximum arity of relations in the signature. Then, a GF-formula φ is satisfiable on standard models iff, for some function $\rho : free(\varphi) \rightarrow V_{LFD}$, $\tau_\rho(\varphi)$ is satisfiable on dependence models.*

Proof. From right to left, if the LFD-formula $\tau_\rho(\varphi)$ is satisfiable, it is also satisfiable (by Proposition 2.9) on a pointed distinguished model $(M, A), s$, and so, by Theorem 3.4 we also have $G(M, A), s \circ \rho \models \varphi$.

From left to right, suppose $M, t \models \varphi$. Since every GF-formula is a Boolean combination of self-guarded formulas, we can assume without loss of generality that t is a guarded assignment. It then follows from Theorem 3.5 together with Theorem 3.4 and the invariance of GF-formulas under GF bisimulations, that, for some map ρ , $\tau_\rho(\varphi)$ is satisfied in a (distinguished) dependence model. \square

Given the compositional nature of our translations τ_ρ on the Boolean operations, this immediately also implies a faithful reduction of semantic consequence in GF to semantic consequence in LFD.

4 A translation from LFD to GF

We now present a satisfiability-preserving translation from LFD to GF. Note that the translation $tr(\cdot)$ from LFD to FO given in Section 2 yields FO formulas that are, in general, not guarded. For example, if $V_{LFD} = \{x, y, z\}$, then $tr(D_x y)$ is the first-order formula $\varphi(x) = \forall y' z' (A(x, y', z') \rightarrow y = y')$, which uses unguarded quantification. Note, though, that the tr -translation of an LFD formula without dependence atoms is guaranteed to be a GF formula. In this section, we present a satisfiability-preserving translation from full LFD, including the dependence atoms, to GF. This translation works by replacing dependence atoms by relational atoms using fresh relations, and adding guarded conditions that force these relations to behave like dependence atoms.¹⁰

Definition 4.1 ($\widehat{\mathbf{S}}$) LFD was defined relative to a relational signature \mathbf{S} and a finite set of variables V_{LFD} . We now denote by $\widehat{\mathbf{S}}$ the relational signature that extends \mathbf{S} with an n -ary relation A , for $n = |V_{LFD}|$, and with a fresh k -ary relation symbol $R^{X,Y}$ for each pair of subsets $X, Y \subseteq V_{LFD}$, where $k = |X|$.

The overall scheme of our translation will look like this:

$$\begin{array}{ccc}
 \text{LFD-formula } \varphi \text{ over } \mathbf{S} & \xrightarrow{\sigma} & \text{GF formula } \sigma(\varphi) \text{ over } \widehat{\mathbf{S}} \\
 \text{dependence model } R(\mathfrak{M}_M) & & \text{standard model } M \\
 \swarrow & & \swarrow \\
 & \text{LFD type model } \mathfrak{M}_M &
 \end{array}$$

In particular, our operation on models takes a detour through type models. In this diagram, R is the mapping from type models to dependence models given by Theorem 2.12.

Next we define the matching formula translation σ .

Definition 4.2 Let $V_{LFD} = \{v_1, \dots, v_n\}$. For an LFD-formula ψ over a signature \mathbf{S} , we define $\sigma(\psi)$ as the following GF-formula over the signature $\widehat{\mathbf{S}}$:

$$\sigma(\psi) := tr^\bullet(\psi) \wedge setup(\psi) \wedge A(v_1, \dots, v_n)$$

where

- $tr^\bullet(\psi)$ is defined inductively in the same way as $tr(\psi)$, with the only difference that $tr^\bullet(D_V U) = R^{V,U}(\bar{v})$. (Here, for the sake of presentation, we assume a fixed, canonical ordering on variables, so that it is clear which element of the tuple \bar{v} corresponds to which element of the set V .)
- $setup(\psi)$ is the conjunction of all GF-sentences of the form

¹⁰This same trick with additional predicates is used in proving the finite dependence model property for LFD by an appeal to Herwig's theorem [12].

$$\begin{aligned} \forall v_1, \dots, v_n (A(v_1, \dots, v_n) \rightarrow \bigwedge_{u \in V} R^{V,u}(\bar{v})) \\ \forall v_1, \dots, v_n (A(v_1, \dots, v_n) \rightarrow (R^{V,U}(\bar{v}) \wedge R^{U,W}(\bar{u}) \rightarrow R^{V,W}(\bar{v}))) \\ \forall v_1, \dots, v_n (A(v_1, \dots, v_n) \rightarrow [R^{V,U}(\bar{v}) \wedge tr^\bullet(\mathbb{E}_V \xi) \rightarrow tr^\bullet(\mathbb{E}_{V \cup U} \xi)]) \end{aligned}$$

where $V, U, W \subseteq V_{LFD}$; $\bar{v}, \bar{u}, \bar{w}$ are canonical enumerations of these sets and $\xi \in Cl(\psi)$ is a formula whose main connective is not conjunction or negation.

The formulas in the “set up” part of the above translation can be viewed as instances of the Projection, Transitivity and Transfer axioms in the axiomatization of LFD given in [4].

Next, we define our operation H , which takes a standard model and produces an LFD type model.

Definition 4.3 Let $V_{LFD} = \{v_1, \dots, v_n\}$ and let ψ be an LFD formula. For a standard model $M \models setup(\psi)$, we define \mathfrak{M}_M^ψ as:

$$\mathfrak{M}_M^\psi := \{\text{type}^\psi(M, s) \mid s : V_{LFD} \rightarrow Dom(M) \text{ such that } M, s \models A(v_1, \dots, v_n)\}$$

with $\text{type}^\psi(M, s) = \{\varphi \in Cl(\psi) \mid M, s \models tr^\bullet(\varphi)\}$, where $Cl(\psi)$ is the closure of φ , as defined in Section 2.2.

Lemma 4.4 *If $M \models setup(\psi)$, then \mathfrak{M}_M^ψ is an LFD type model.*

Proof. First, we show that each $\Delta \in \mathfrak{M}_M^\psi$ is a type. The \neg -consistency, \wedge -consistency, and \mathbb{E} -consistency properties are immediate from the construction of \mathfrak{M}_M^ψ and the definition of tr^\bullet . Projection and transitivity hold by virtue of the corresponding conjuncts of $setup(\psi)$.

The *universal* property, i.e., that \sim_\emptyset is the universal relation on \mathfrak{M} , also follows immediately from the construction of \mathfrak{M}_M^ψ using the fact that tr^\bullet -translation of an LFD-sentence is a GF-sentence.

Finally, for the *witness* property, suppose that $\mathbb{E}_V \varphi \in \Delta$. Let s be a witnessing assignment for Δ . Then, in particular, $M, s \models A(v_1, \dots, v_n) \wedge tr^\bullet(\mathbb{E}_V \varphi)$. It follows by the definition of tr^\bullet that there is an assignment t with $t =_V s$, such that $M, t \models A(v_1, \dots, v_n) \wedge tr^\bullet(\varphi)$. Let $\Delta' \in \mathfrak{M}_M^\psi$ be the type obtained from t as in Definition 4.3. Then $\varphi \in \Delta'$. Moreover, since $t =_V s$, and the translation tr^\bullet preserves the free variables of the formula, we have that $\Delta \sim_V \Delta'$. In order to satisfy the witness property of type models, we must show something slightly stronger, namely that $\Delta \sim_{D_V^\Delta} \Delta'$. This can be shown in two steps: (i) it is easy to see that $D_V^{\Delta'} = D_V^\Delta$ by virtue of the fact that $\text{free}(tr^\bullet(D_V U)) = V$; (ii) the third conjunct of $setup(\psi)$ now allows us to lift the $\Delta \sim_V \Delta'$ relation to the $\Delta \sim_{D_V^\Delta} \Delta'$. To see this, first note that, by the \neg -consistency and \wedge -consistency properties of types, it is enough to show that Δ and Δ' agree on “non-decomposable” formulas with free variables in D_V^Δ , where we call a formula decomposable if its main connective is a conjunction or a negation. Let $\xi \in \Delta$ with $Y = \text{free}(\xi) \subseteq D_V^\Delta$, and assume that ξ is non-decomposable. Then $M, s \models tr^\bullet(\xi)$ and $M, s \models R^{V,Y}(\bar{v})$. By existential

generalisation (cf. the definition of tr^\bullet) we get that $M, s \models tr^\bullet(\mathbb{E}_V \xi)$. Since $s =_V t$, this implies $M, t \models tr^\bullet(\mathbb{E}_V \xi)$ and $M, t \models R^{V,Y}(\bar{v})$ as well. Since we know that $M, t \models setup(\psi)$, we can now apply the ‘transfer’ condition to get that $M, t \models tr^\bullet(\mathbb{E}_{V \cup Y} \xi)$ which implies that $M, t \models \xi$ as $free(\xi) \subseteq V \cup Y$. Hence $\xi \in \Delta'$. The converse direction follows from a symmetric argument. \square

We now obtain our dependence model $H(M)$ by applying the unraveling operation R from Section 2.2 to \mathfrak{M}_M^ψ .

Definition 4.5 Fix an LFD formula ψ over signature \mathbf{S} , let M be a standard model for $\widehat{\mathbf{S}}$ s.t. $M \models setup(\psi)$, and let $s : V_{LFD} \rightarrow dom(M)$ such that $M, s \models A(v_1, \dots, v_n)$.

- (i) We define $H(M)$ as $R(\mathfrak{M}_M^\psi)$.
- (ii) We define $H(s)$ as an (arbitrarily chosen) admissible LFD assignment on $H(M)$ that realizes $type^\psi(M, s)$ in $H(M)$.

While not reflected in our notation, the model $H(M)$ depends on ψ . Also, note that $type^\psi(M, s) \in \mathfrak{M}_M^\psi$, and, by Theorem 2.12, $H(M)$ realizes every type in \mathfrak{M}_M^ψ , making $H(s)$ indeed well-defined.

Theorem 4.6 Fix an LFD formula ψ over signature \mathbf{S} , and let M be standard model over the signature $\widehat{\mathbf{S}}$ such that $M \models setup(\psi)$, and let $s : V_{LFD} \rightarrow Dom(M)$, such that $M, s \models A(v_1, \dots, v_n)$. Then, for all formulas $\varphi \in Cl(\psi)$,

$$H(M), H(s) \models \varphi \quad \text{iff} \quad M, s \models tr^\bullet(\varphi)$$

Proof. If $H(M), H(s) \models \varphi$, then by the definition of $H(s)$, we have that $\varphi \in type^\psi(M, s)$, i.e., $M, s \models tr^\bullet(\varphi)$. Conversely, if $M, s \models tr^\bullet(\varphi)$, then $\varphi \in type^\psi(M, s)$. Hence, again by the definition of $H(s)$, we have that $H(M), H(s) \models \varphi$. \square

Corollary 4.7 (i) An LFD formula ψ is satisfiable iff its GF-translation $\sigma(\psi)$ is satisfiable.

- (ii) An LFD entailment $\varphi \models \psi$ is valid iff the GF-entailment $setup(\varphi) \wedge tr^\bullet(\varphi) \wedge A(v_1, \dots, v_n) \models setup(\psi) \rightarrow tr^\bullet(\psi)$ is valid.

Proof. From left to right, if $(M, A), s \models \psi$, then let M' be the $\widehat{\mathbf{S}}$ -expansion of M obtained by setting $A^{M'}$ as $\{(s(v_1), \dots, s(v_n)) \mid s \in A\}$, and each $(R^{V,U})^{M'} = \{s(\bar{v}) \mid (M, A), s \models D_V U(\bar{v})\}$. A straightforward formula induction shows that, for all LFD formulas φ , $(M, A), s \models \varphi$ iff $M', s \models A(v_1, \dots, v_n) \wedge tr^\bullet(\varphi)$. Furthermore, $M' \models setup(\psi)$ since all auxiliary predicates have their intended interpretation. Therefore, we have that $M', s \models \psi$. The converse direction follows immediately from Theorem 4.6.

(ii) follows from (i), because (as careful inspection shows) $setup(\varphi \wedge \neg\psi)$ is logically equivalent to $setup(\varphi) \wedge setup(\psi)$ (to see this, note that, if $\varphi \wedge \neg\psi$ has a subformula that is non-decomposable, then the latter must be a subformula of φ or of ψ) and the fact that tr^\bullet commutes with the Boolean connectives. \square

5 Transfer results from our translations

In addition to shedding light on the relationship between LFD and GF, our translations can also be used to transfer results from GF to LFD and vice versa. We give several examples connecting known results, but also a new result, viz. a complete complexity analysis of LFD.

As a warming up, we reprove a known result, namely the finite model property for GF [7]. Recall that a logic has the finite model property (FMP) if every satisfiable formula has a finite model. It has been long known that GF has the FMP [7], and FMP was only recently shown to hold for LFD [12]. Therefore, the following transfer fact does not give us a new result, but it does provide an illustration of the value of our translations.

Theorem 5.1 (FMP for GF) *GF has the finite model property.*

Proof. Let φ be a satisfiable guarded formula. By corollary 3.6, since φ is satisfiable on standard models, it follows that $\tau_\rho(\varphi)$ is satisfiable on dependence models, for some $\rho : \text{free}(\varphi) \rightarrow V_{LFD}$ (for V_{LFD} sufficiently large). By the finite model property of LFD [12], there is a finite dependence model \mathbb{M} satisfying $\tau_\rho(\varphi)$ at some $s \in A$. We can take \mathbb{M} to be distinguished by Proposition 2.9 (which preserves finiteness). In fact, inspection of the proof in [12] shows that its construction via Herwig’s theorem already ensures distinguishedness. By Theorem 3.4, then, $G(\mathbb{M})$ is a finite model of φ . \square

Theorem 5.2 (Complexity of LFD satisfiability) *For a finite set of LFD-variables V_{LFD} , the satisfiability problem for LFD-formulas in V_{LFD} is ExpTime-complete. The same problem is 2ExpTime-complete if V_{LFD} is considered as part of the input.*

Proof. Upper bound: by reduction to GF. The satisfiability problem for GF is 2ExpTime-complete [7]. More precisely, the satisfiability of a GF formula with k FO-variables can be checked in time $2^{O(|\varphi| \cdot k^k)}$ (cf. [14]). A careful analysis of our translation from LFD to GF shows it can be performed in time $O(|\varphi|) \cdot 2^{O(k)}$, where $k = |V_{LFD}|$. Indeed, computing the modified first-order translation $tr^\bullet(\varphi)$ can be done in linear time, so only the set-up part brings all the complexity. The formula $setup(\varphi)$ consist of 2^k many conjuncts encoding a couple of projection axiom instances on the proxy-dependence atoms $R^{X,y}$ as well as 2^{3k} many conjuncts encoding a transitivity axiom instance and $O(|\varphi|) \cdot 2^{2k}$ many conjuncts encoding a transfer axiom instance for some ‘indecomposable’ formulas in the closure $Cl(\varphi)$. Putting this together, we obtain that the satisfiability problem for LFD is in ExpTime if V_{LFD} is treated as fixed in the complexity analysis, and that it is in 2ExpTime if V_{LFD} is part of the input (due to the fact that, even though the translation from LFD to GF is exponential, it does not increase the number of variables).

Lower bound: by reduction from GF. Observe that our translation from GF to LFD is exponential due to the disjunction in the translation clause for the guarded quantifiers. Exponential in the nesting depth of (polyadic) guarded-quantifications, to be precise. For this reason, it is not immediately clear that

complexity upper bounds for GF transfer to LFD as is. However, we can make use of a result from [7], according to which there is a satisfiability-preserving translation from GF-formulas to a “Scott-normal form” (which can be performed in polynomial time and without increasing the number of variables), where the normal form uses only two levels of polyadic guarded quantification. Our translation is polynomial when applied to GF-formulas in this normal form. As we mentioned earlier, the satisfiability problem for GF-formulas with a bounded number of variables is ExpTime-complete, and with an unbounded number of variables it is 2ExpTime-complete. Since our translation from GF to LFD does not require more LFD variables than the number of variables in the input formula, this establishes our lower bounds. \square

In fact, we can improve this result a little. By the *monadic fragment* of LFD we mean LFD-formulas that only use unary relation symbols.

Theorem 5.3 (Complexity of the monadic fragment of LFD) *Fix a finite set V_{LFD} of LFD-variables with $|V_{LFD}| \geq 2$. The satisfiability problem for the monadic fragment of LFD with V_{LFD} is ExpTime-complete.*

Proof. [sketch] The upper bound follows from Theorem 5.2. For the lower bound, we reduce from the satisfiability problem of the basic modal logic \mathbf{K} extended with the global modality (which we will denote as $\mathbf{K} + U$). This logic is known to be ExpTime-complete (cf. [6]). By a simple encoding trick due to Halpern and Vardi [10], the satisfiability problem for $\mathbf{K} + U$ reduces to the satisfiability problem for the multi-modal logic $\mathbf{S5}_2 + U$ (where $\mathbf{S5}_2$ is the bi-modal “fusion” logic that has two S5 modality without interaction axioms). The coding trick in question consists of replacing every occurrence of \diamond by $\diamond_1 \diamond_2$ where \diamond_1 and \diamond_2 are the two $\mathbf{S5}$ -modalities (cf. [10] for the proof that this preserves satisfiability). The satisfiability problem for $\mathbf{S5}_2 + U$, finally, embeds straightforwardly into the monadic fragment of LFD (even without using dependence atoms): each proposition letter becomes a unary predicate, \diamond_1 becomes \mathbb{E}_x and \diamond_2 becomes \mathbb{E}_y for $x, y \in V_{LFD}$ two distinct LFD-variables, whereas the global modality (in its existential form) becomes \mathbb{E}_\emptyset . \square

Remark 5.4 Our translation from GF to (the dependence-atom-free fragment of) LFD can also be composed with the *tr* translation from LFD to GF to obtain a SAT-reduction from GF to its “universal guard” fragment (i.e., GF-formulas with a single guard predicate that occurs only in guard position).¹¹

As our final example, we prove that LFD has Craig interpolation, offering a model-theoretic alternative to the sequent calculus-based proof in [4].

Theorem 5.5 (Craig interpolation for LFD) *For every valid LFD implication $\models \varphi \rightarrow \psi$, there is an LFD formula ϑ such that*

- (i) $\models \varphi \rightarrow \vartheta$,
- (ii) $\models \vartheta \rightarrow \psi$, and

¹¹In fact, to its fragment where all quantifiers are guarded by the same atom $A(v_1, \dots, v_n)$.

(iii) *all relation symbols occurring in ϑ occur both in φ and in ψ .*

Proof. [sketch] Let $\varphi \rightarrow \psi$ be valid in LFD. By Corollary 4.7, the implication $(\text{setup}(\varphi) \wedge \text{tr}^\bullet(\varphi) \wedge A(v_1, \dots, v_n)) \rightarrow ((\text{setup}(\psi) \wedge A(v_1, \dots, v_n)) \rightarrow \text{tr}^\bullet(\psi))$ is valid in GF. Note that, in this GF formula, all quantification is guarded by A . Furthermore, the antecedent and the negation of the consequent are self-guarded (meaning that the free variables are guarded by an atomic conjunct), and the free variables of the consequent form a subset of the free variables of the antecedent (viz. $\{v_1, \dots, v_n\}$). Hence, by the weak Craig interpolation theorem [11, Thm. 4.5], there is a GF-formula χ such that

- (i) $\models (\text{setup}(\varphi) \wedge \text{tr}^\bullet(\varphi) \wedge A(v_1, \dots, v_n)) \rightarrow \chi$,
- (ii) $\models \chi \rightarrow ((\text{setup}(\psi) \wedge A(v_1, \dots, v_n)) \rightarrow \text{tr}^\bullet(\psi))$, and
- (iii) All relation symbols occurring in χ except possibly for A , occur both in $(\text{setup}(\varphi) \wedge \text{tr}^\bullet(\varphi))$ and in $((\text{setup}(\psi) \wedge A(v_1, \dots, v_n)) \rightarrow \text{tr}^\bullet(\psi))$.
- (iv) The free variables of χ belong to V_{LFD} .

It suffices only to translate χ back to LFD. This is not entirely trivial, since χ is now a GF-formula over the expanded signature $\hat{\mathbf{S}}$. That is, it will in general contain the A and $R^{V,U}$ relations. Fortunately, it turns out that the translation τ from GF to LFD that we gave in Section 3 can be extended in a straightforward way to the full signature $\hat{\mathbf{S}}$, as follows:

$$\begin{aligned} \tau_\rho(A(\bar{x})) &= \top \text{ if } \rho(\bar{x}) \text{ is precisely the sequence } v_1, \dots, v_n; \perp \text{ otherwise} \\ \tau_\rho(R^{U,V}(\bar{u})) &= D_{\rho[U]\rho[V]} \end{aligned}$$

For any dependence model $\mathbb{M} = (M, A)$, let $\hat{\mathbb{M}}$ be the standard model that is the $\hat{\mathbf{S}}$ -expansion of M obtained in the natural way (as we did in the proof of Corollary 4.7). Then, it can be shown that, with the above modification to τ ,

- (a) For all distinguished \mathbb{M} and admissible s , and for all GF-formulas ξ over $\hat{\mathbf{S}}$, we have $\hat{\mathbb{M}}, s \circ \rho \models \xi$ iff $\mathbb{M}, s \models \tau_\rho(\xi)$.
- (b) For every LFD formula ξ , $\tau_\rho(\text{tr}^\bullet(\xi))$ is equivalent to ξ over distinguished dependence models, and hence over all dependence models, if we take ρ to be the identity map on V_{LFD} .

Putting this all together, it follows that $\vartheta := \tau_\rho(\chi)$ (under the above modified translation function τ , and with ρ the identity map on V_{LFD}) is a Craig interpolant for our original implication $\varphi \rightarrow \psi$ on distinguished dependence models, and hence, by Proposition 2.9, on all dependence models. \square

LFD also has Craig interpolants that only use shared variables, [4]. To obtain this further information, the preceding analysis should be refined.

6 Further Directions

Language extensions One natural question is if our results extend to richer languages than the ones considered here. Here is an example.

The translation τ_ρ from GF into LFD presented in Section 3 can be extended to deal with *identity atoms* $x = y$ between GF variables x, y . It suffices to add two clauses: (a) $\tau_\rho(x = y) = \top$ if $\rho(x) = \rho(y)$ [the mapping to LFD variables enforces identity throughout], (b) $\tau_\rho(x = y) = \perp$ if $\rho(x) \neq \rho(y)$ [the values will always be distinct in distinguished dependence models]. Thus GF(=), too, translates compositionally into the decidable logic LFD without identity.

Another natural extension from an LFD perspective is adding *local independence atoms* $I_X y$ saying that fixing the local values of the variables X at assignment s in the current dependence model puts no constraint on y : which can still take any value in its range on the X -restricted subset of assignments. Adding atoms $I_X y$ to LFD results in an undecidable logic [4]. Still, even from the original motivation for CRS, analyzing true FO quantifiers in terms of their independence behavior versus the dependence behavior of CRS quantifiers makes sense, suggesting a study of richer fragments of FO.¹²

Translation patterns Our three translations show some general patterns.

Translating LFD into the FO language via $tr(\cdot)$ was compositional, while there was an inverse model transformation T supporting the usual contravariant translation equivalence $M, s \models tr(\varphi)$ iff $T(M), s \models \varphi$. Thus there is an adjunction between the maps $tr(\cdot), T$ whose transfer of properties between models has been determined in an abstract setting in [17]. In fact, T was onto and bijective, which allows for additional transfer of properties between LFD and FO to the extent that LFD can be identified with a fragment of first-order logic.

The compositional translation τ_ρ from GF into LFD also came with an inverse model transformation G supporting a contravariant equivalence, but this time, it only worked for the special classes of distinguished LFD models and correlated distinguished GF models. However, all models for LFD were LFD-bisimilar to these distinguished models, and an analogous result held for GF. This relaxed notion of translation up to bisimulation for the relevant two languages seems an interesting generalization of the standard case which still facilitates a good deal of transfer. We saw this for logical consequence, it also works straightforwardly for decidability, and we even saw how, putting together the translation with its matching model transformation allowed us to transfer the finite model property.

Finally, our translation from LFD into GF was not entirely compositional, as we also needed to carry special conditions for LFD type models for the formula being translated. This amounts to translating from LFD on its ordinary models into GF on a special class of models satisfying a theory consisting of special conditions for type models. This theory allowed for a transformation of its GF models into LFD type models, which can then, in a modular fashion, be represented as ordinary LFD dependence models.

Several general questions arise here. One is about the range of the third type of translation technique. We believe that it can deal quite generally with

¹²Other natural language extensions to explore would introduce *fixed-point operators*, [7], a device whose power has not yet been studied for the dependence logic LFD.

modal-style logics that have an effective ‘finite quasimodel property’ plus a representation theorem.¹³ We leave this matter for further exploration.¹⁴

7 Conclusion

The guarded fragment GF models restricted quantification, the modal dependence logic LFD models local dependence between variables. We have presented two new translation results. One is a faithful compositional translation from GF into the dual CRS fragment of LFD, the other is an effective reduction from full LFD with dependence atoms to GF. We then demonstrated a number of consequences for transfer of known properties between GF and LFD, and also derived some new results, such as a determination of the computational complexity of satisfiability in LFD. In summary, local dependence and guarding are much closer as semantic notions than what may have been thought previously. Moreover, the techniques that we introduce to prove our results may be of wider interest, and we provided some pointers in our final discussion.

References

- [1] Abiteboul, S., R. Hull and V. Vianu, “Foundations of Databases,” Addison-Wesley, 1995. URL <http://webdam.inria.fr/Alice/>
- [2] Andréka, H., I. Németi and J. van Benthem, *Modal languages and bounded fragments of predicate logic*, Journal of Philosophical Logic **27** (1998).
- [3] Andréka, H., J. van Benthem, N. Bezhanishvili and Í. Németi, “Changing a Semantics: Opportunism or Courage?” Birkhäuser Verlag, Basel, 2007 pp. 307–337.
- [4] Baltag, A. and J. van Benthem, *A simple logic of functional dependence*, J. Philos. Log. **50** (2021), pp. 939–1005. URL <https://doi.org/10.1007/s10992-020-09588-z>
- [5] Barany, V., B. ten Cate and L. Segoufin, *Guarded negation*, Journal of the ACM (JACM), Association for Computing Machinery (2015), pp. 22:1–22:26.
- [6] Blackburn, P., M. de Rijke and Y. Venema, “Modal Logic,” Cambridge University Press, Cambridge UK, 2001.
- [7] Grädel, E., *On the restraining power of guards*, Journal of Symbolic Logic **64** (1998), pp. 1719–1742.
- [8] Grädel, E. and M. Otto, “Guarded Teams: The Horizontally Guarded Case,” Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2020 pp. 22:1–22:17.
- [9] Grädel, E. and P. Pützstück, *Logics of dependence and independence: The local variants*, Journal of Logic and Computation **31** (2021), pp. 1690–1715.
- [10] Halpern, J. Y. and M. Y. Vardi, *The complexity of reasoning about knowledge and time. i. lower bounds*, Journal of Computer and System Sciences **38** (1989), pp. 195–237. URL <https://www.sciencedirect.com/science/article/pii/002200089900391>

¹³As an illustration, consider the modal logic K4 which is complete for transitive models. We cannot translate this directly into GF since transitivity is not guarded. This can be overcome by suitably translating into the guarded fixed-point logic $\mu(\text{GF})$, but a simpler solution is this. Translate the requirements on finite filtrations for K4 models, where in particular, if $\Box\varphi$ belongs to a type, then both $\Box\varphi, \varphi$ belong to accessible types. The result is a guarded description of filtration models.

¹⁴A further interesting issue would be an abstract general transfer analysis of the generalized types of translation that we have provided in this paper.

- [11] Hoogland, E. and M. Marx, *Interpolation and definability in guarded fragments*, *Studia Logica* **70** (2002), pp. 373–409.
- [12] Koudijs, R., *Finite model property and bisimulation for lfd*, *Electronic Proceedings in Theoretical Computer Science* **346** (2021), pp. 166–178.
URL <http://dx.doi.org/10.4204/EPTCS.346.11>
- [13] Marx, M., *Taming first order logic: Relating the semantic and the syntactic approach*.
URL <https://festschriften.illc.uva.nl/j50/contribs/marx/index.html>
- [14] Marx, M. and Y. Venema, “Local Variations on a Loose Theme: Modal Logic and Decidability,” Springer Berlin Heidelberg, Berlin, Heidelberg, 2007 pp. 371–429.
- [15] Németi, Í., “Free algebras and decidability in algebraic logic,” Doctoral Dissertation for the Hungarian Academy of Sciences, 1986, translation of the relevant chapter as: Decidability of weakened versions of first-order logic. In *Logic Colloquium 92*, Stanford, CSLI Publications, 1995. pp.177–241.
- [16] Väänänen, J., “Dependence Logic,” Cambridge University Press, Cambridge UK, 2009.
- [17] van Benthem, J., *Information transfer across chu spaces*, *Logic Journal of the IGPL* **6** (2000), pp. 719–731.
- [18] van Benthem, J., *Guards, bounds, and generalized semantics*, *J. Log. Lang. Inf.* **14** (2005), pp. 263–279.
URL <https://doi.org/10.1007/s10849-005-5786-y>
- [19] Van Benthem, J., “Crs and Guarded Logics,” *Bolyai Society Mathematical Studies* **22**, Mathematical Institute, Hungarian Academy of Sciences, Budapest, 2012 pp. 273 – 301.