Bridging the Gap: Intelligent Environments with Smart Materials

Hedda R. Schmidtke
Institute of Information Systems
Universität zu Lübeck
Lübeck, Germany
schmidtke@ifis.uni-luebeck.de

Mena Leemhuis
Institute of Information Systems
Universität zu Lübeck
Lübeck, Germany
leemhuis@ifis.uni-luebeck.de

Jana Mertens
Mechatronic Systems Lab
Technische Universität Berlin
Berlin, Germany
jana.mertens@tu-berlin.de

Robert Courant
Mechatronic Systems Lab
Technische Universität Berlin
Berlin, Germany
robert.courant@tu-berlin.de

Jürgen Maas
Mechatronic Systems Lab
Technische Universität Berlin
Berlin, Germany
juergen.maas@tu-berlin.de

Özgür L. Özcep
Institute of Information Systems
Universität zu Lübeck
Lübeck, Germany
oezcep@ifis.uni-luebeck.de

***Abstract*—Smart Materials (SMat) promise to open new opportunities in the area of Intelligent Environments (IE), whether as part of dedicated smart devices or as the fabric constituting everyday appliances and building infrastructure. Through the use of ontologies both IE engineers and the IEs themselves can be aware of, and predict, how novel configurable and changing materials react under different conditions. In contrast to conventional Smart Objects, however, as computational software/hardware-systems, lending themselves to the object-oriented perspective of conventional ontology specification languages, SMat and IE in the wider sense require a perspective focussing on extended spaces and numerical domains. Both are known to be problematic in terms of usability and computational complexity for the traditional object-oriented languages, with even very basic notions already leading into undecidability. Context Logic (CL), in contrast, is a formalism specialized for these domains. This paper demonstrates how terminology from this area involving extended spaces and numerical domains can be modeled in CL.**

***Index Terms*—ontology, smart materials, OBDA, context logic, spatial granularity**

I. Introduction

Smart Materials (SMat) promise to open new opportunities in the area of Intelligent Environments (IE). Future materials offering sensor and actuator functionality are not only embeddable in dedicated smart devices and wearables but can become a fabric for everyday appliances or building materials [1], [2]. A key obstacle to realizing this vision is the communication and knowledge gap between engineers and systems in the SMat and IE domains. Ontologies have been proposed as a general bridging technology [3]. However, knowledge about advanced materials comes in a number of formats not easily incorporated into ontologies [4]–[7]. A particularly challenging class of knowledge about materials can be called function-related knowledge, descriptions of how a material's parameters change in dependence on other parameters being changed. Such knowledge can be stored, e.g., as CSV files with recorded experimental data or simulation data, or as complex equations given, e.g., in matrix format or in terms of a partial differential equation. The ontological task at hand, in particular, towards reliable IE [8] is to facilitate querying and leveraging such information both by IE engineers and the systems themselves, so as to, e.g., be able to predict, protect from, and react to, critical system states, such as a forgotten oven in an Ambient Assisted Living scenario.

Within the field of knowledge representation, function-related knowledge has been discussed particularly with respect to processes and change over time [9]–[13]. Formally and technically, this type of information is available, e.g., in time series data and time-dependent parameterized equations describing physical properties of materials under given environmental circumstances. Especially, changes in space or location with respect to the geographic domain have received considerable attention [14]. For the well-studied case of motion of a rigid object but also more generally, one can distinguish two main approaches: a) an *object-oriented* approach, in which a three-dimensional object is described as undergoing a change of location, i.e., as having a different location property at different times [9]; and b) a *four-dimensionalist* (generalized: multi-dimensionalist) approach, in which not a three-dimensional object is considered at given times, but its four-dimensional trace, i.e., a four-dimensional region (for the philosophical perspective cf. [15]). This *field-based* view is vital for representing knowledge in the physical sciences [13], [16].

The latter approach views functions mathematically as a special type of relation, and relations again as subsets or regions in a multi-dimensional coordinate space. It has a number of advantages, in particular, in terms of generality, e.g.: we need not distinguish between essential and accidental properties, can conceptually rotate, project, or slice a given point set as desired and handle conceptual granularity. On the side of knowledge representation and ontology engineering

The authors are grateful for funding by BMBF within the project SmaDi, grant number 13XP5124A/B.

experts, in contrast, the *object-based* view, as used, e.g., in OWL, the ontology standard of the Semantic Web, is more common. OWL and its underlying formal framework of Description Logics (DL) [17] rely on a class/concept-oriented framework emphasizing the strict separation into objects, their properties, and classes. DL's underlying semantics relying on sets of discrete objects for class semantics and pairs of discrete objects for property semantics, renders representing continuous domains such as regions and value ranges difficult. These domains – called "concrete" domains in the DL area – have been proven intractable within the framework [4]–[7].

A related aspect is the integration of knowledge from relational databases (RDBs). An approach proposed to handle this issue with OWL is ontology-based data access (OBDA) [18]. OBDA operates on external data sources within the RDB paradigm by translating ontological queries, e.g., in SPARQL, into SQL queries. Logically, OBDA is supported by being based on rewriting queries in the first-order logical (FOL) framework underlying SQL. In order to enable rewriting into FOL, the logic used to represent the terminology is restricted to the family of lightweight logics DL-Lite [19]. This family, in turn, is the basis for the W3C recommended profile OWL 2 QL[1], which is a strict fragment of full OWL 2.

These restrictions limit the usefulness towards a more general Semantic Web, especially for the physical sciences. For this purpose, it would be desirable to leverage a logical language for which the field-based perspective is a first class citizen, i.e., that does not require translation. A promising candidate language is Context Logic (CL). Developed specifically for handling sensory and spatiotemporal continuous domains with a multi-dimensionalist perspective and based on a mereological rather than set-theoretical semantics [20]–[24], CL makes describing function-related knowledge and spatiotemporal entities easy while also providing basic support for specifying subclass hierarchies. A Hilbert-style granular mereogeometry [25] showed that the language's specific mereological perspective allows it to specify a geometry with the same representational flexibility in dimensional variation as known from topology, allowing it to serve as a facilitator for bringing function-related knowledge into the Semantic Web. This paper demonstrates for examples from a concrete knowledge base consisting of a textually given ontology, currently partially encoded in DL, accompanied by, inter alia, CSV files containing characteristic curve data, how knowledge about smart materials can be encoded in CL.

The rest of the article is structured as follows. We start by sketching the knowledge base and data set focussing on two representative examples (Section II) and presenting the syntax and semantics of CL (Section III). We show how the knowledge would be represented in CL (Section IV) and discuss example queries and their processing within the general reasoning infrastructure. The paper closes with a discussion of open questions and future work (Section V). A decidable reasoner for the central fragment of the language is outlined in the appendix.

[1]\https://www.w3.org/TR/owl2-profiles/#OWL_2_QL

II. Use Case

While a detailed exposition of the dielectric elastomer use case is beyond the scope of this paper, it is worth remarking that the material sciences examples below are part of a larger use case comprising of, so far: 253 axioms generated from linguistically provided expert domain knowledge and a range of external data sources, including tabular information, such as CSV-files specifying, e.g., characteristic curves, as well as scripts and formulae capable of generating tabular information given input parameters.

We focus on two particular examples.

Example 1. *A crucial procedure in material science is the tensile test. For one of the smart materials in the project, this knowledge is provided as a CSV-file of two recorded measurement parameters, from which three further parameters are computed via external scripts.*

The crucial challenge in this example is the embedding of numerical data from CSV-files and external scripts. A conventional OBDA-environment requires adherence to the OWL2QL restrictions, and numerical data (concrete domains) are beyond the expressiveness of this language.

A second restriction of OWL2QL is, that it does not permit concepts involving an existential quantification that contains concept-restrictions containing themselves an existential quantification. This capability is necessary in the second example of a dielectric elastomer stack-transducer [26], [27]:

Example 2. *A multilayered dielectric elastomer stack-transducer (DETs) is a layering of several DE transducers. Transducers based on dielectric elastomers (DEs) consist of a polymer as dielectric between two compliant electrodes and can convert electrical into mechanical energy and vice versa.*

The second example is problematic for OWL2QL because it would require a DET to be specified as a layering of complex parts. This is similar to a standard counterexample not covered in OWL2QL of a grandparent, which is someone who has a child that is a parent, with parent, as a person who has a child, being itself a concept involving quantification.[2]

III. Context Logic

Context Logics are a cognitively motivated family of languages. This focus was originally chosen in order to overcome usability issues of DLs in the area of ontology-based context-aware environments. In this IE-scenario, system administrators should on a day to day basis be capable of adjusting IE-system behaviors through logical statements but require little or no expertise in ontology languages. The key idea was that by being close to the fundamental structures of human cognition and language, a logical language hierarchy could be designed so that fragments with lower computational complexity are

[2]For further details on DLs cf. [17], a more detailed discussion is beyond the scope of this article.

components within the languages of higher complexity. The aim was to facilitate fast reasoning for cognitively, linguistically, and logically simple statements, while clearly guarding statements potentially leading to longer reasoning times in an intuitively accessible manner. If what is difficult for the reasoner is difficult for the human user, it is easier for the user to intuit the machine's behavior.

In terms of their underlying theory, the CL languages are, on the one hand, based on the evolutionary hierarchy of levels of detachment of reference in cognition and language proposed by Gärdenfors [28], [29]. On the other hand, they are motivated by the mereological approach to representing knowledge about the world [11], [15], [20], [30]–[32]. A key characteristic of the language family is that they form a hierarchy that follows the evolutionary cognitive hierarchy in both syntactic structure and computational complexity for the reasoning task. Its highest layer, called layer-1 or first order context logic (CL1) is syntactically identical to a standard mereological first-order logic (FOL) framework. In terms of the wider landscape in formal semantics of logical languages, CL has recently been found to be most closely related to Fuzzy Logics as being based on formal lattice semantics as well as featuring a related analogous semantics [33]–[36].

A main novelty of CL, in contrast to FOL, lies in its capability to further analyze the basic predicate expressions of FOL in terms of a more primitive language: atomic CL (CLA, [35], [37]–[40]). CLA is similar in syntax and reasoning complexity to propositional logic or monadic first order logic and corresponds linguistically to simple predication sentences. In between the two languages lies the quantifier-free, layer-0, propositional, or 0th order context logic (CL0, [33], [41]), which allows the specification of alternatives.

A. Syntax

CL is syntactically a two-layered language with a term layer and a formula layer. On the first layer, *context terms* $\mathcal{T}_C$ are defined over a set of variables $\mathcal{V}_C$:

- TY1 Any context variable $v \in \mathcal{V}_C$ and the special symbols $\top$ and $\bot$ are atomic context terms.
- TY2 If c is a context term, then its complement $(\sim c)$ is a context term.
- TY3 If c and d are context terms then the intersection $(c \sqcap d)$ and sum $(c \sqcup d)$ are context terms.

On the second layer, *context formulae* $\mathcal{F}_C$ are defined as follows:

- FY1 If c and d are context terms then $[c \sqsubseteq d]$ is an atomic context formula.
- FY2 If ϕ is a context formula and c is a context term, then $(\neg\phi)$ and $c : \phi$ are context formulae.
- FY3 If ϕ and ψ are context formulae then $(\phi \wedge \psi)$, $(\phi \vee \psi)$, $(\phi \rightarrow \psi)$, and $(\phi \leftrightarrow \psi)$ are context formulae.
- FY4 If $x \in \mathcal{V}_C$ is a context variable and ϕ is a context formula, then $\forall x : \phi$ and $\exists x : \phi$ are context formulae.

In the following, we leave out brackets as far as possible applying the precedence: $\sim, \sqcap, \sqcup$ for term operators and $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$ for formula operators. The scope of quantifiers is to be read as maximal, i.e., until the first bracket closes that was opened before the quantifier, or until the end of the formula. Square brackets around atomic formulae are used for easier visual separation between term layer and formula layer. We denote the subset of variables of $\mathcal{V}_C$ occurring in a given formula ϕ as $\mathcal{V}_C(\phi)$.

A strict hierarchy of three fragments can be distinguished alone on the basis of the formula operators allowed. We thus obtain the CL hierarchy:

- **CLA** *Atomic* CL comprises of all formulae that can be constructed with FY1, contextualization in FY2, and $\wedge$ in FY3.
- **CL0** *Level-0* CL comprises of all formulae that can be constructed with FY1-FY3.
- **CL1** *Level-1* CL comprises of all formulae that can be constructed with FY1-FY4.

B. Semantics

Different variant semantics have been proposed including a possible world semantics [41], and a semantics in terms of a labelled deductive system [42]. The different approaches slightly differ in the resulting semantics, but all employ a lattice structure for specifying the meanings of context terms, assigning a partial order to give a semantics to $\sqsubseteq$. We here provide a categorical semantics that unites these different perspectives under the umbrella of topology.

We characterize assignment as a functor category $\mathcal{A} = [\mathcal{V}_C, \mathcal{O}]$, where $\mathcal{O}$ is a topology over a space $\mathcal{U}$ with $\rightarrowtail$ symbolizing the maps between the objects of $\mathcal{O}$. $\mathcal{V}_C$ is the set of context variables. With an intended lattice semantics in mind [33], $\rightarrowtail$ is the lattice partial order, i.e., transitive, reflexive, and antisymmetric, with sum ($\oplus$) and meet ($\odot$) the lattice operators, with bounds $\mathcal{U}$ and $\emptyset$ added for convenience.

The semantics needs to include an interpretation for complex context terms in $\mathcal{T}_C$. Since this interpretation depends solely on the assignments $a \in \mathcal{A}$ for the variables in $t \in \mathcal{T}_C$, we leverage the canonical extension $\mathcal{C} \subseteq [\mathcal{T}_C, \mathcal{O}]$ from $\mathcal{A}$. For any $a \in \mathcal{A}$, the derived $c \in \mathcal{C}$ have the following properties:

- TS1 For all $v \in \mathcal{V}_C$: $c(v) = a(v)$, and $c(\top) = \mathcal{U}$, $c(\bot) = \emptyset$.
- TS2 For any $\gamma \in \mathcal{T}_C$: $c(\sim\gamma) = \bigoplus_{o \in \mathcal{O}, (o \odot c(\gamma)) = \emptyset} o$.
- TS3 If $c(\gamma_1) = o_1$ and $c(\gamma_2) = o_2$, then $c(\gamma_1 \sqcap \gamma_2) = o_1 \odot o_2$ and $c(\gamma_1 \sqcup \gamma_2) = o_1 \oplus o_2$.

We can see that if $\mathcal{O}$ is the special case of a Boolean algebra, we receive the usual propositional logic semantics. At the same time, we can then interpret the result with respect to the Boolean algebra topology as the set-theoretic variant of propositional logic.

With an interpretation for context terms, we obtain a semantics for CL0, context logics without quantification, as $\mathcal{M} \subseteq \mathcal{O} \times \mathcal{C} \times \mathcal{L}_C$. We can proceed with formulae as:

- FS1 For any context terms $\alpha_1, \alpha_2 \in \mathcal{T}_C$:
  - a) $m = (o, c, \alpha_1 \sqsubseteq \alpha_2) \in \mathcal{M}$ iff $o \odot c(\alpha_1) \rightarrowtail c(\alpha_2)$ is a map in $\mathcal{O}$.
- FS2 For any formula ϕ and context term α:

a) $m = (o, c, \neg\phi) \in \mathcal{M}$ iff $m' = (o, c, \phi) \notin \mathcal{M}$
b) $m = (o, c, \alpha : \phi) \in \mathcal{M}$ iff $m' = (o \odot c(\alpha), c, \phi) \in \mathcal{M}$

FS3 For formulae ϕ and ψ:
a) $m = (o, c, \phi \wedge \psi) \in \mathcal{M}$ iff $m_1 = (o, c, \phi) \in \mathcal{M}$ and $m_2 = (o, c, \psi) \in \mathcal{M}$
b) $m = (o, c, \phi \vee \psi) \in \mathcal{M}$ iff $m_1 = (o, c, \phi) \in \mathcal{M}$ or $m_2 = (o, c, \psi) \in \mathcal{M}$
c) $m = (o, c, \phi \rightarrow \psi) \in \mathcal{M}$ iff $m_1 = (o, c, \phi) \notin \mathcal{M}$ or $m_2 = (o, c, \psi) \in \mathcal{M}$

A problem with logical systems allowing arbitrary sums and intersections is that we may want to be able to talk about such entities without taking them under consideration when quantifying. We can meaningfully say, for instance, that "Pegasus is a horse with wings" without wanting to commit to "there is a horse with wings." On the contrary, we may want to add explicitly: "but horses with wings do not exists." A simple way to do this is to restrict quantification to a subset of entities of $\mathcal{O}$. For adding quantification, we therefore need a means to separate valuation functions $\mathcal{E}$ to entities to which we want to ontologically commit and quantify over from the valuation functions in $\mathcal{C}$ by using only a subset $\mathcal{E} \subseteq \mathcal{C}$ for quantification. Quantification can thus be restricted to certain entities. Accordingly, a model of CL1 is characterized by interpreting quantifiers with respect to $\mathcal{E}$.

FS4 $m = (o, c, \exists x : \phi) \in \mathcal{M}$ iff there is a $c' \in \mathcal{E}$ that agrees with c in all regards except potentially for $c(x)$ so that $(o, c', \phi) \in \mathcal{M}$ holds.

FS5 $m = (o, c, \forall x : \phi) \in \mathcal{M}$ iff for all $c' \in \mathcal{E}$ that agree with c in all regards except potentially for $c(x)$ holds $(o, c', \phi) \in \mathcal{M}$.

The logic of atomic context logic formulae (CLA) corresponds to propositional logic. A main benefit of CL is the ability to use the contextualization syntax without, e.g., having to assume distinct world indices as in Hybrid Logics [43]. But from a point of view of expressiveness, contextualization does not extend the language beyond what FS1 already provides (for the proof cf. [36], [38]) since

$$\gamma : [\alpha \sqsubseteq \beta] \equiv [\alpha \sqcap \gamma \sqsubseteq \beta]$$

and for complex ϕ, contextualization in $c : \phi$ distributes down to the atomic formula level independent of logical operator or quantification. Consequently, contextualization could be defined syntactically.

C. Decidable Fragments

Two fragments of CL, atomic CL (CLA) and propositional CL (CL0) are known to be decidable (cf. Section A and [41]), and many representation and reasoning tasks can be performed in CL0. The key difference between CLA and CL0 from a mereological perspective, is that CL0 has logical negation ($\neg$) and thus allows the specification of overlap ($\bigcirc$) between entities

$$[a \bigcirc b] \overset{def}{\Leftrightarrow} \neg[a \sqcap b \sqsubseteq \bot] \tag{1}$$

with further derivable relations *non-empty part of* ($\dot{\sqsubseteq}$) and *proper part of* ($\sqsubset$)

$$[a \mathrel{\dot{\sqsubseteq}} b] \overset{def}{\Leftrightarrow} [a \sqsubseteq b] \wedge [a \bigcirc b], \tag{2}$$

$$[a \sqsubset b] \overset{def}{\Leftrightarrow} [a \sqsubseteq b] \wedge \neg[b \sqsubseteq a]. \tag{3}$$

Overlap allows an ontology engineer to specify arbitrary finite graphs in CL0 by describing the *connection* between entities using *overlap* and a connecting element that allows separate entities to be connected. Generally speaking, we can describe any given finite graph as usual as consisting of nodes connected by edges, where edges can be – to a user invisible – abstract anonymous entities, in directed graphs with edges as pairs, with a distinct first and second part. However, edges and their distinct parts can also be meaningful and may linguistically be referred to.

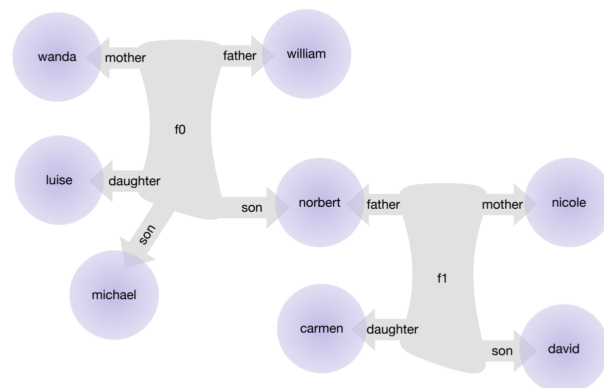

Fig. 1. An individual, Norbert, as a member of two generations of the Smith family: $[f_0 \sqsubseteq smith] \wedge [f_0 \sqcap norbert \sqsubseteq son] \wedge \neg[f_0 \sqcap norbert \sqsubseteq \bot] \wedge [f_1 \sqsubseteq smith] \wedge [f_1 \sqcap norbert \sqsubseteq father] \wedge \neg[f_1 \sqcap norbert \sqsubseteq \bot]$.

Figure 1 shows such an example. It shows two generations of the Smith family characterized mereologically via linguistically plausible edge constructions f_0 and f_1 belonging to the social domain. We see that these edges correspond to core families, i.e., meaningful entities and minimal sub-families of the Smith family. The larger family would be the graph consisting of the sum of the components f_i. We can specify a number of derived concepts like *parent* and *child* with CLA alone to add more structure to the description of edges.

$$[parent = mother \sqcup father] \tag{4}$$

$$[child = son \sqcup daughter] \tag{5}$$

At the first glance, specifying secondary relations, such as *grandmother of* or *father of* seems to need quantification, i.e., CL1. This would require leaving the decidable fragment CL0:

$$father(x, y) \overset{def}{\Leftrightarrow} \exists e : [x \sqcap e \mathrel{\dot{\sqsubseteq}} father] \wedge [y \sqcap e \mathrel{\dot{\sqsubseteq}} child] \tag{6}$$

$$grandmother(x, y) \overset{def}{\Leftrightarrow} \exists e_0 : [x \sqcap e_0 \mathrel{\dot{\sqsubseteq}} mother] \wedge \exists p : [p \sqcap e_0 \mathrel{\dot{\sqsubseteq}} child] \wedge \exists e_1 : [p \sqcap e_1 \mathrel{\dot{\sqsubseteq}} parent] \wedge [y \sqcap e_1 \mathrel{\dot{\sqsubseteq}} child] \tag{7}$$

Formally, this construction is a *tuple generator*, with which arbitrary relations can be constructed. Note, however, that when used as schemata, with schema variables filled by known context variables, we can simply instantiate the existentially quantified entities e, e_0, e_1, p with newly introduced context variables in the manner of the fundamental Skolemization procedure. All of the formulae above contain, apart from the schema variables on the left hand side of the definition, only existentially quantified variables on the right hand side. Applying the definition from left to right is, thus, unproblematic.

From the point of view of conventional ontology design, this capability is crucial as it allows the specification of assertional knowledge about entities, thus, contributing to bridging the gap between the object-oriented and mereological perspective. We can, thus, in particular, specify the instance-of relation between an object o and a class c:

$$\exists e : [e \sqsubseteq isi] \wedge [o \sqcap e \dot{\sqsubseteq} a_1] \wedge [c \sqcap e \dot{\sqsubseteq} a_2]. \tag{8}$$

Generalizing, we can write any type of relational constructions in the conventional manner of standard FOL via such Context Logic schemata. We can, moreover, distinguish between partial orders and more general relations. We would describe a variant of the temporal interval relation *starts* [44] widely employed in IE reasoning as a preorder constructed from two fundamental preorders c and t providing different aspects of temporal ordering [36] with the quantifier-free schema (9). The above *is instance of* can be generalized with the CL1-schema (10). The construction also allows arbitrary arity as would be needed, e.g., for *between* (11):

$$starts(\alpha, \beta) \overset{def}{\Leftrightarrow} c : [\alpha \sqsubseteq \beta] \wedge t : [\alpha \sqsubseteq \beta], \tag{9}$$

$$isi(\alpha, \beta) \overset{def}{\Leftrightarrow} \exists e : [e \sqsubseteq isi] \wedge [\alpha \sqcap e \dot{\sqsubseteq} a_1] \wedge [\beta \sqcap e \dot{\sqsubseteq} a_2], \tag{10}$$

$$btw(\alpha, \beta, \gamma) \overset{def}{\Leftrightarrow} \exists e : [e \sqsubseteq btw] \wedge [\alpha \sqcap e \dot{\sqsubseteq} a_1] \wedge [\beta \sqcap e \dot{\sqsubseteq} a_2] \wedge [\gamma \sqcap e \dot{\sqsubseteq} a_3]. \tag{11}$$

In ontologies, we can leverage an abbreviation syntax with square and round brackets to indicate the two types of relational constructs, with square brackets indicating a relation is a partial order, thus further shortening definitions in ontologies:

$$x[\alpha, \beta] \overset{def}{\Leftrightarrow} x : [\alpha \sqsubseteq \beta]. \tag{12}$$

We thus have derived the conventional syntax of *predicate expressions* demonstrating that these can be considered internally complex constructions with CL. We gain the advantage of reducing the number of axioms required for basic and compound preorder relations, such as the above *starts*. Such derived preorders, thus, automatically gain their powerful transitive reasoning properties in CL without requiring any axioms.

IV. Representing the Examples

We can now show how the knowledge from the use case examples can be conceptualized and represented in CL. We start with the numerical values of example 1.

A. Numerical Values with Units

Given knowledge about numerical values, we can characterize function values via $\sqcap$ as regions. Our notion is the same as in a 2D numerical graph (Figure 2). The depiction of a specific $f(x)$-value can be understood as the intersection between two regions. While in functional analysis graphs are

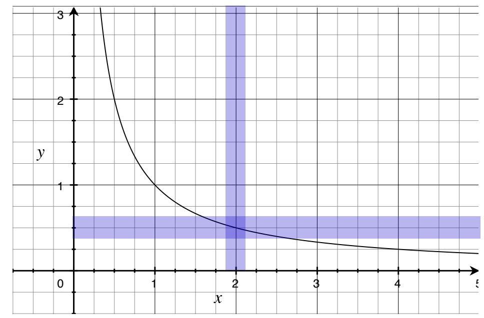

Fig. 2. A mereological conceptualization of points on a graph

usually drawn for two or three numerical dimensions, the same idea is used as well in bar charts and grouped bar charts, with less restrictions on data types proving general applicability. For material science, functional graphs are key components of knowledge. Characteristic curves, such as shown in Figure 3, describe a material's behavior as input values or environmental conditions change.

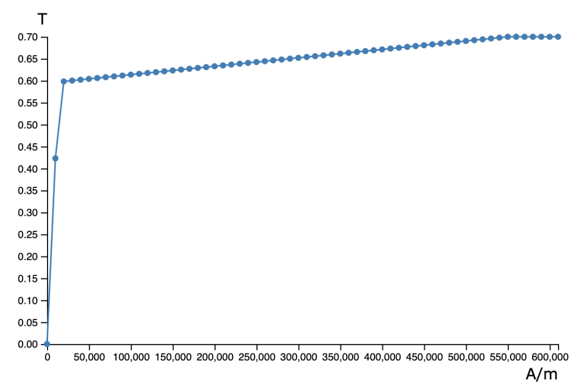

Fig. 3. Example of a characteristic curve showing magnetic field strength plotted against magnetic polarization for a sample, as extracted and visualized by our system.

B. Spatial Relations and Granularity

In the second example, we need to specify a DET as a layering of complex DEs, a case that resembles the above grandparent example. We saw above that this example has ontologically two interesting aspects: first, the formal specification via edge connections and second the fact that examples were apparently in the more complex fragment CL1, but unproblematic in practical application.

For the former aspect, we see that the spatial mereological approach is particularly intuitive with respect to the material science domain. Whereas the family connection was an abstract construction, physical relations require local interaction: in our example 2, domain experts have concrete answers for the question as to what binds certain objects together, locally. The abstract notion of granularity employed in knowledge representation frameworks with size-based granularity [45] provides a means to formally represent the various physical types of local spatial interaction between materials. Thus to qualitatively specify the relation between layers of films in a CL ontology, we can simply leverage knowledge about the material connection by which components are attached to each other in physical reality and obtain a qualitative but accurate characterization.

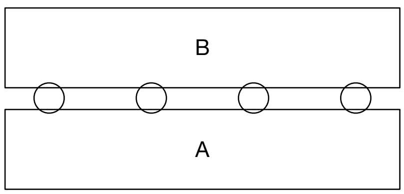

Fig. 4. A notion of separable adjacency in size-based granularity [45]: A and B are separate objects but both overlap some spatial locations of (for a context of given physical interaction principles) irrelevant extension.

The notion of *irrelevant extension* in Figure 4 is a notion of size-based granularity. The actual size range of what is *irrelevant extension*, i.e., what proximity is meant as constituting the physical relation, is a crucial component of the material description. A physical material connection, e.g.: a covalent bond connecting materials on the level of molecular orbitals is stronger than a hook-and-loop fastener at the millimeter level; a polymer dispersion adhesive operating at the molecular level can provide adhesive strength at the intermediate level. The actual physical reality of a given case determines which type of situation is given.

Without an axiomatic characterization of the natural numbers, such as the Peano axioms, well-known to lead to undecidability, we cannot describe a stack or a DET, in general. We can, however, describe an n-film DET, for which we know the number of films n, via an inductive characterization:

$$subclass : [ts_1 = t\_film] \quad (13)$$

$$isi(x, ts_{n+1}) \overset{def}{\Leftrightarrow} \exists y, z : isi(y, ts_n) \wedge isi(z, t\_film) \wedge pcnc(y, z) \quad (14)$$

where *isi* is the relation *instance of* defined above, and *pcnc* is a notion of *planar connectedness* [25], [45]. Given that each occurrence of a relation symbol (here: *isi* and *pcnc*) expands into another existential quantification, the schema involves three more existentially quantified variables, i.e., a sum of five existential quantifiers. The schema itself has a schema variable x, which in FOL or CL1 could be understood as universally quantified. For any given x, however, we can generate its complete graph of components simply by introducing new context variables for any of the five existentially quantified variables, in a manner similar to the Skolemization procedure in FOL reasoning. For any given x specified as $isi(x, ts_{n+1})$, i.e., in application from left to right, all subsequent reasoning thus remains in CL0. When querying whether a given entity is a DET, moreover, we can understand the existentially quantified variables as query variables.

V. Conclusions

Smart Materials open a range of new opportunities for future Intelligent Environments. Specifying their behavior under different environmental conditions with ontologies is as with more classical sensor-actuator systems, vital to their predictable and safe application. We suggested the usage of the mereologically founded Context Logic as bringing numerous advantages over traditional object-oriented languages widely used in ontology engineering. While object-oriented languages suggest a worldview of objects, classes, and slot-filler roles, CL offers an inherently spatial perspective suitable to describe multidimensional fields, spaces, and space-time frameworks at different levels of granularity. We demonstrated for the representative examples of a characteristic curve and the definition of dielectric elastomer stack-transducers, how reasoning tasks not covered by current OBDA approaches can be handled in CL. A range of further questions regard the connection to object-oriented frameworks. Here, future works should, *inter alia*, focus on further facilitating interoperability with DL.

References

[1] H. Janocha, Ed., *Adaptronics and smart structures*. Springer, 2007.
[2] M. Shahinpoor, *Fundamentals of Smart Materials*. Royal Society of Chemistry, 2020.
[3] N. Guarino, "Formal ontology and information systems," in *Formal Ontology in Information Systems*, N. Guarino, Ed. Amsterdam: IOS Press, 1998, pp. 3–15.
[4] G. Kahn and G. D. Plotkin, "Concrete domains," *Theoretical Computer Science*, vol. 121, no. 1-2, pp. 187–277, 1993.
[5] V. Haarslev, C. Lutz, and R. Möller, "A description logic with concrete domains and a role-forming predicate operator," *Journal of Logic and Computation*, vol. 9, no. 3, pp. 351–384, 1999.
[6] C. Lutz, "Reasoning with concrete domains," in *IJCAI*. Citeseer, 1999, pp. 90–95.
[7] S. Demri and K. Quaas, "Concrete domains in logics: a survey," *ACM SIGLOG News*, vol. 8, no. 3, pp. 6–29, 2021.
[8] J. C. Augusto and A. Coronato, "Introduction to the inaugural issue of the journal of reliable intelligent environments," *Journal of Reliable Intelligent Environments*, vol. 1, no. 1, pp. 1–10, 2015. [Online]. Available: https://doi.org/10.1007/s40860-015-0005-3
[9] J. Allen, "Towards a general theory of action and time," *Artificial Intelligence*, vol. 23, pp. 123–154, 1984.
[10] A. Galton, "Space, time and movement," in *Spatial and Temporal Reasoning*. Kluwer, 1997, pp. 321–352.
[11] B. Bennett, "Space, time, matter and things," in *Proceedings of the 2nd international conference on Formal Ontology in Information Systems (FOIS'01)*, C. Welty and B. Smith, Eds. Ogunquit: ACM, 2001, pp. 105–116.
[12] J. Pustejovsky, J. M. Castano, R. Ingria, R. Sauri, R. J. Gaizauskas, A. Setzer, G. Katz, and D. R. Radev, "Timeml: Robust specification of event and temporal expressions in text." *New directions in question answering*, vol. 3, pp. 28–34, 2003.
[13] A. Galton, "Fields and objects in space, time, and space-time," *Spatial Cognition and Computation*, vol. 4, no. 1, pp. 39–68, 2004.
[14] ——, "Space, time, and the representation of geographical reality," *Topoi*, vol. 20, no. 2, pp. 173–187, 2001.
[15] W. V. O. Quine, P. S. Churchland, and D. Føllesdal, *Word and object*. MIT press, 2013.

[16] H. Couclelis, "People manipulate objects (but cultivate fields): Beyond the raster-vector debate in GIS," in *Spatio-Temporal Reasoning*, A. U. Frank, I. Campari, and U. Formentini, Eds., 1992, pp. 65–77.
[17] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, Eds., *The description logic handbook: Theory, implementation and applications*. Cambridge University Press, 2003.
[18] A. Poggi, D. Lembo, D. Calvanese, G. D. Giacomo, M. Lenzerini, and R. Rosati, "Linking data to ontologies," *Journal of Data Semantics*, vol. 10, pp. 133–173, 2008.
[19] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, A. Poggi, M. Rodríguez-Muro, and R. Rosati, "Ontologies and databases: The DL-Lite approach," in *Semantic Technologies for Informations Systems – 5th Int. Reasoning Web Summer School (RW 2009)*, S. Tessaris and E. Franconi, Eds. Springer, 2009, pp. 255–356.
[20] B. Smith, "Mereotopology: A theory of parts and boundaries," *Data and Knowledge Engineering*, vol. 20, pp. 287–303, 1996.
[21] I. Pratt and O. Lemon, "Ontologies for plane, polygonal mereotopology," *Notre Dame Journal of Formal Logic*, vol. 38, no. 2, pp. 225–245, 1997.
[22] A. C. Varzi, "Basic problems of mereotopology," in *Formal Ontology in Information Systems*. IOS Press, 1998, pp. 29–38.
[23] A. G. Cohn, "Mereotopology," in *Encyclopedia of GIS*, S. Shekhar and H. Xiong, Eds. Springer, 2008, p. 652.
[24] S. Borgo and C. Masolo, "Full mereogeometries," *The review of symbolic logic*, vol. 3, no. 04, pp. 521–567, 2010.
[25] H. R. Schmidtke, "Granular mereogeometry," in *Formal Ontology in Information Systems: Proceedings of the 9th International Conference (FOIS 2016)*, ser. Frontiers in Artificial Intelligence and Applications, R. Ferrario and W. Kuhn, Eds., vol. 283. IOS Press, 2016, pp. 81–94.
[26] A. Hubracht, O. Çabuk, T. Krüger, and J. Maas, "Fabrication and characterization of thin film-based multilayered dielectric elastomer stack transducers via submodules," in *Electroactive Polymer Actuators and Devices (EAPAD) XXIII*, I. A. Anderson, H. R. Shea, and J. D. W. Madden, Eds., vol. 11587, International Society for Optics and Photonics. SPIE, 2021, p. 115870U. [Online]. Available: https://doi.org/10.1117/12.2584768
[27] J. Mertens, O. Cabuk, and J. Maas, "Modeling and experimentally parameterizing approach of the viscoelastic material behavior of DE-based transducer materials," in *Electroactive Polymer Actuators and Devices (EAPAD) XXIV*, I. A. Anderson, J. D. W. Madden, and H. R. Shea, Eds., vol. 12042, International Society for Optics and Photonics. SPIE, 2022, p. 1204214. [Online]. Available: https://doi.org/10.1117/12.2613368
[28] P. Gärdenfors, *Language and the evolution of cognition*. Lund University, 1995.
[29] P. Gärdenfors, *The Dynamics of Thought*. Springer, 2005.
[30] J. J. Srzednicki and Z. Stachniak, Eds., *Leśniewski's Systems Protothetic*, ser. Nijhoff International Philosophy Series. Springer Netherlands, 2012, vol. 54.
[31] P. Simons, *Parts – A Study in Ontology*. Oxford: Oxford University Press, 1987.
[32] R. Casati and A. C. Varzi, *Parts and Places: the Structure of Spatial Representations*. MIT Press, 1999.
[33] H. R. Schmidtke, "Towards a fuzzy context logic," in *Fuzzy Systems - Theory and Applications*, C. Volosencu, Ed. IntechOpen, 2021.
[34] P. Hájek, *Metamathematics of fuzzy logic*, ser. Trends in Logic. Springer Science & Business Media, 1998, vol. 4.
[35] H. R. Schmidtke, "Logical lateration – a cognitive systems experiment towards a new approach to the grounding problem," *Cognitive Systems Research*, vol. 52, pp. 896 – 908, 2018.
[36] H. R. Schmidtke and S. Coelho, "Scales and hedges in a logic with analogous semantics," in *Ninth Annual Conference on Advances in Cognitive Systems*, 2022, pp. 1–20.
[37] H. R. Schmidtke and M. Beigl, "Distributed spatial reasoning for wireless sensor networks," in *Modeling and Using Context*. Springer, 2011, pp. 264–277.
[38] H. R. Schmidtke, "Contextual reasoning in context-aware systems," in *Workshop Proceedings of the 8th International Conference on Intelligent Environments*. IOS Press, 2012, pp. 82–93.
[39] ——, "Textmap: A general purpose visualization system," *Cognitive Systems Research*, vol. 59, pp. 27–36, 2020.
[40] ——, "Reasoning and learning with Context Logic," *Journal of Reliable Intelligent Environments*, vol. 7, no. 2, pp. 171–185, 2021.
[41] H. R. Schmidtke, D. Hong, and W. Woo, "Reasoning about models of context: A context-oriented logical language for knowledge-based context-aware applications," *Revue d'Intelligence Artificielle*, vol. 22, no. 5, pp. 589–608, 2008.
[42] H. R. Schmidtke, H. Yu, P. Masomo, A. Kinai, and A. Shema, "Contextual reasoning in an intelligent electronic patient leaflet system," in *Context in Computing: A Cross-disciplinary Approach for Modeling the Real World*, P. Brézillon and A. Gonzalez, Eds. Springer, 2014, pp. 557–573.
[43] P. Blackburn, "Representation, reasoning, and relational structures: a hybrid logic manifesto." *Logic Journal of the IGPL*, vol. 8, no. 3, 2000.
[44] J. F. Allen, "Maintaining knowledge about temporal intervals," *Communications of the ACM*, vol. 26, no. 11, pp. 832–843, 1983.
[45] H. R. Schmidtke and W. Woo, "A size-based qualitative approach to the representation of spatial granularity," in *Twentieth International Joint Conference on Artificial Intelligence*, M. M. Veloso, Ed., 2007, pp. 563–568.

APPENDIX

We define a simple tableaux reasoning mechanism for CL0 based upon the characterization of a reasoning mechanism for a similar language [41] and sketch proofs of soundness and completeness. A detailed discussion of practical issues is beyond the scope of this article.

Informally, a tableau reasoner works by expanding sets of formulae, called tableaux, according to semantically founded rules, either until no further expansion is possible, that is, the tableaux are *saturated*, or until a contradiction that cannot be avoided is found. In the latter case, the algorithm yields "unsatisfiable"; otherwise, the algorithm answers "satisfiable," and the resultant set of tableaux can be interpreted as a model fulfilling the asked question. The reasoning mechanism can be defined as follows.

a) Definition: a *CL-tableau* t is a set of formulae $t = \Gamma \cup \Delta$ over a finite set of context variables $\Sigma \subseteq \mathcal{V}_C$, where the set Γ contains the positive formulae, that is those formulae whose outer operator is not $\neg$, whereas Δ contains the negative formulae, that is, those formulae whose outer operator is $\neg$. A tableau is *saturated* iff it has the following properties for complex formulae:

$$\phi \wedge \psi \in \Gamma \text{ then } \phi \in \Gamma \text{ and } \psi \in \Gamma \tag{T1}$$
$$\neg(\phi \wedge \psi) \in \Delta \text{ then } \neg\phi \in \Delta \text{ or } \neg\psi \in \Delta \tag{T2}$$
$$\phi \vee \psi \in \Gamma \text{ then } \phi \in \Gamma \text{ or } \psi \in \Gamma \tag{T3}$$
$$\neg(\phi \vee \psi) \in \Delta \text{ then } \neg\phi \in \Delta \text{ and } \neg\psi \in \Delta \tag{T4}$$
$$\phi \rightarrow \psi \in \Gamma \text{ then } \neg\phi \in \Delta \text{ or } \psi \in \Gamma \tag{T5}$$
$$\neg(\phi \rightarrow \psi) \in \Delta \text{ then } \phi \in \Gamma \text{ and } \neg\psi \in \Delta \tag{T6}$$
$$\neg\neg\phi \in \Delta \text{ then } \phi \in \Gamma \tag{T7}$$

and complex context terms:

$$[c \sqsubseteq d] \in \Gamma \text{ then } \neg[\top \sqsubseteq c] \in \Delta \text{ or } [\top \sqsubseteq d] \in \Gamma \tag{T8}$$
$$[\top \sqsubseteq {\sim}c] \in \Gamma \text{ then } [c \sqsubseteq \bot] \in \Gamma \tag{T9}$$
$$\neg[\top \sqsubseteq {\sim}c] \in \Delta \text{ then } \neg[c \sqsubseteq \bot] \in \Delta \tag{T10}$$
$$[\top \sqsubseteq c \sqcap d] \in \Gamma \text{ then } [\top \sqsubseteq c] \in \Gamma \text{ and } [\top \sqsubseteq d] \in \Gamma \tag{T11}$$
$$\neg[\top \sqsubseteq c \sqcap d] \in \Delta \text{ then } \neg[\top \sqsubseteq c] \in \Delta \text{ or } \neg[\top \sqsubseteq d] \in \Delta \tag{T12}$$
$$[\top \sqsubseteq c \sqcup d] \in \Gamma \text{ then } \neg[c \sqsubseteq \bot] \in \Delta \text{ or } \neg[d \sqsubseteq \bot] \in \Delta \tag{T13}$$

A tableau is called *disjoint* iff there is no formula $\phi \in \Gamma$, so that $\neg\phi \in \Delta$. The saturation rules alone do not yield a complete reasoning mechanism for context logics, as two cases are not yet covered.

b) Definition:: a Hintikka system in Context Logics is a pair $H = (T, P)$ where P is a reflexive, transitive, and antisymmetric relation over T, a non-empty set of disjoint, saturated tableaux fulfilling the following three properties. For tableaux $t = \Gamma \cup \Delta, t' = \Gamma' \cup \Delta'$ with $t, t' \in T$ and $P(t, t')$ holds that

$$\text{for all atomic formulae } \phi\text{: } \phi \in \Gamma \text{ implies } \phi \in \Gamma'. \tag{H1}$$

For any tableau $t = \Gamma \cup \Delta$ with $\neg[c \sqsubseteq d] \in \Delta$, there is a tableau $t' = \Gamma' \cup \Delta'$ with $P(t, t')$ such that

$$[\top \sqsubseteq c] \in \Gamma' \text{ and } \neg[\top \sqsubseteq d] \in \Delta'. \tag{H2}$$

For any tableau $t = \Gamma \cup \Delta$ with $\neg[\top \sqsubseteq c \sqcup d] \in \Delta$, there is a tableau $t' = \Gamma' \cup \Delta'$ with $P(t, t')$ such that

$$[c \sqsubseteq \bot] \in \Gamma' \text{ and } [d \sqsubseteq \bot] \in \Gamma'. \tag{H3}$$

$H = (T, S)$ is a CL-Hintikka system for a tableau $t = \Gamma \cup \Delta$ if there is a tableau $t' = \Gamma' \cup \Delta'$ in T such that $\Gamma \subseteq \Gamma'$ and $\Delta \subseteq \Delta'$.

The question whether a query sentence κ follows from a context knowledge base CKB can be formulated as the question whether the tableau $t = CKB \cup \{\neg\kappa\}$ has a Hintikka system.

c) Proof (⇒, soundness): We follow the convention and write $(M, o) \models \phi$ to abbreviate $(o, c, \phi) \in \mathcal{M}$ for some c. We provide a 1:1-mapping that establishes the connection between models $M \subseteq \mathcal{O} \times \mathcal{C} \times \mathcal{L}$ and Hintikka systems $H = (T, P)$:

$$T = \{\Gamma_x \cup \Delta_x \mid \forall \phi \in \mathcal{L}_\Sigma : \phi \in \Gamma_x \text{ iff } (M, x) \models \phi, \text{ and } \neg\phi \in \Delta_x \text{ iff } (M, x) \not\models \phi\} \tag{16}$$

$$P((\Gamma_x \cup \Delta_x), (\Gamma_y \cup \Delta_y)) \text{ iff } x \rightarrowtail y \tag{17}$$

We show that (T, P) actually is a Hintikka system, i.e., that the tableaux $(\Gamma_x \cup \Delta_x) \in T$ thus defined are disjoint and saturated and that the conditions (H1)-(H3) hold. Disjointness follows from the fact that, for any $x \in \mathcal{O}$ and any formula $\phi \in \mathcal{L}_\Sigma$, either $(M, x) \models \phi$ and $\phi \in \Gamma_x$, or $(M, x) \not\models \phi$ and $\neg\phi \in \Delta_x$.

That each t is saturated can be seen by checking the correspondence between each of the saturation rules and the definition of the semantics. The case of $[c \sqsubseteq d] \in \Gamma_x$ (T8), in particular, demands that $\neg[\top \sqsubseteq c] \in \Delta_x$ or $[\top \sqsubseteq d] \in \Gamma_x$. This is given, since the corresponding semantics ensures that $(M, x) \models [c \sqsubseteq d]$ iff for all y with $x \rightarrowtail y$, $(M, y) \not\models [\top \sqsubseteq c]$ or $(M, y) \models [T \sqsubseteq d]$. Since $x \rightarrowtail x$ (reflexivity of $\rightarrowtail$) this also holds for x and thus we also have either $\neg[\top \sqsubseteq c] \in \Delta_x$, or $[\top \sqsubseteq d] \in \Gamma_x$. Similarly, we can directly show saturation for $[\top \sqsubseteq c]$ and $[\top \sqsubseteq c \sqcap d]$, whether they are in Γ or negated in Δ, and saturation for the case $[\top \sqsubseteq c \sqcup d] \in \Gamma$, as well as saturation for formulae constructed using the propositional logic connectives.

Also, the rules (H1)-(H3) hold for H. The first rule follows from the semantics of $[c \sqsubseteq d]$ together with the transitivity of $\rightarrowtail$: if $[c \sqsubseteq d] \in \Gamma_x$ and thus $(M, x) \models [c \sqsubseteq d]$, then for all y with $x \rightarrowtail y$: $(M, y) \not\models [\top \sqsubseteq c]$ or $(M, y) \models [\top \sqsubseteq d]$. Since $x \rightarrowtail y$ and $y \rightarrowtail y'$ entails $x \rightarrowtail y'$ for all y', it follows that also $(M, y') \not\models [\top \sqsubseteq c]$ or $(M, y') \models [\top \sqsubseteq d]$ and thus $(M, y) \models [c \sqsubseteq d]$ and $[c \sqsubseteq d] \in \Gamma_y$. For the second rule, assume $(\Gamma_x \cup \Delta_x) \in T$ with $\neg[c \sqsubseteq d] \in \Delta_x$. In this case, $(M, x) \not\models [c \sqsubseteq d]$ holds. Then we know that there is $y \in \mathcal{O}$ with $x \rightarrowtail y$ and $(M, y) \models [\top \sqsubseteq c]$ and $(M, y) \not\models [\top \sqsubseteq d]$; and accordingly, there must be a tableau $(\Gamma_y \cup \Delta_y) \in T$ with $P((\Gamma_x \cup \Delta_x), (\Gamma_y \cup \Delta_y))$, $[\top \sqsubseteq c] \in \Gamma_y$, and $\neg[\top \sqsubseteq d] \in \Delta_y$. For the third rule, we analogously assume $(\Gamma_x \cup \Delta_x) \in T$ with $\neg[\top \sqsubseteq c \sqcup d] \in \Delta_x$. This entails that $(M, x) \not\models [\top \sqsubseteq c \sqcup d]$ and thus that there is $y \in \mathcal{O}$ with $x \rightarrowtail y$ and $(M, y) \models [c \sqsubseteq \bot]$ and $(M, y) \models [d \sqsubseteq \bot]$. The corresponding tableau $(\Gamma_y \cup \Delta_y) \in T$ with $P((\Gamma_x \cup \Delta_x), (\Gamma_y \cup \Delta_y))$ fulfills the requirement $[c \sqsubseteq \top] \in \Gamma_y$ and $[d \sqsubseteq \top] \in \Gamma_y$.

d) Proof (⇐, weak completeness): we show that the procedure indicated by the rules terminates after finitely many steps, with the model as indicated above, by induction on the number of occurrences of context variables $varN : \mathcal{L}_C \to \mathbb{N}$ in a formula as a measure of formula complexity. The basis of the induction are the formulae $\phi = [\top \sqsubseteq \alpha]$ with $varN = 1$ for context variables α or $varN = 0$ for constants $\top$ and $\bot$. In this case, there is nothing to do and the procedure terminates. That the two properties hold in this case immediately follows as $t \rightarrowtail a(\alpha)$ entails $(M, t) \models [\top \sqsubseteq \alpha]$. Similarly, for the negated formula $\neg\phi$. For formulae ϕ of the shapes $[c \sqsubseteq d]$ and $[\top \sqsubseteq c \sqcup d], [\top \sqsubseteq c \sqcap d]$ with arbitrarily complex context terms c, d, the procedure follows the structure of the context terms, with $varN(\phi) = varN(c) + varN(d)$. We assume the property holds for all $n < varN(\phi)$.

For $[c \sqsubseteq d]$, first suppose $[c \sqsubseteq d] \in \Gamma$ for a tableau $t = \Gamma \cup \Delta$. By the saturation rules, $\neg[\top \sqsubseteq c] \in \Delta$ or $[\top \sqsubseteq d] \in \Gamma$. And by the first rule, this property holds also for all tableau $t' = \Gamma' \cup \Delta'$ for which $t \rightarrowtail t'$ holds. By the induction assumption, this implies $(M, t') \not\models [\top \sqsubseteq c]$ or $(M, t') \models [\top \sqsubseteq d]$ for all t' with $t \rightarrowtail t'$. And this in turn implies by the semantics that $(M, t) \models [c \sqsubseteq d]$. If $\neg[c \sqsubseteq d] \in \Delta$, then by the second rule there is a tableau $t' = \Gamma' \cup \Delta'$ with $t \rightarrowtail t'$ such that $[\top \sqsubseteq c] \in \Gamma'$ and $\neg[\top \sqsubseteq d] \in \Delta'$. By the induction assumption, this entails that $(M, t') \models [\top \sqsubseteq c]$ and $(M, t') \not\models [\top \sqsubseteq d]$ and thus that $(M, t) \not\models [c \sqsubseteq d]$ by the definition of the semantics. For the cases of $[\top \sqsubseteq \sim c]$ and $[\top \sqsubseteq c \sqcap d]$ in Γ or negated in Δ, the saturation rules directly correspond to the semantics definition. While *varN* is not reduced in one step in the case of $[\top \sqsubseteq \sim c]$ – the rules reduce to $[c \sqsubseteq \bot]$, which is of the shape $[c \sqsubseteq d]$ discussed above – the reduction of *varN* continues after finitely many steps. The case of $[\top \sqsubseteq c \sqcup d] \in \Gamma$ (or negated in Δ) is analogous to that of $[c \sqsubseteq d] \in \Gamma$ (negated in Δ, respectively).

For formulae constructed using the propositional connectives $\neg, \wedge, \vee, \rightarrow$, the reasoning procedure reduces formula complexity with every step until the atomic formula level is reached. □