

Representing Parametric Concepts with Situation Theory

Roussanka Loukanova

Department of Mathematics, Stockholm University

Email: rloukanova@gmail.com

Abstract—We use higher-order, type-theoretic Situation Theory to model semantic concepts as situation-theoretic objects consisting of parametric information. Situation Theory contributes by representing concepts as classes of parametric objects, in a computational way. We use concepts that are often expressed by human language in taxonomy classifications, as a demonstration of the situation theoretic-approach to model parametric information in abstract concepts.

I. INTRODUCTION

THE IDEAS of Situation Theory were originally introduced by Barwise [1], and then by Barwise and Perry [2], for modeling information in nature. The work emerged from decades of efforts by varieties of model-theoretic approaches for adequate computational semantics of human language and cognitive science. In search for adequate semantics of human language, with his extensive work in mathematics, model-theory, and admissible sets, Jon Barwise soon realized that semantic objects for human language are special cases of objects in a more general theory of meaning and information. Since then, Situation Theory has been under development as a powerful, highly expressive theory of finely-grained information that is partial, underspecified, and situational. Semantics of languages is one of the prominent applications of Situation Theory, know as Situation Semantics.

Our intensive efforts on modeling semantic information and concepts are currently in several, concurrent directions, for intelligent applications to information and language processing. In nature, information typically is partial, parametric, and dependent on situations, in most of its components. For adequate modeling of semantic objects, we need to represent these natural features of information and languages. Applications to many contemporary technologies, which are related to data science, information, and language processing, require models of information and information processing. These models need to reflect information flow in nature and, in the same time, to be computational.

On the side of the mathematical foundations of Situation Theory, Aczel non-well-founded set theory, see Aczel [3], has proved to be the most suitable set theory for modeling classes of situation-theoretic objects that are proper classes, i.e., which are not sets in the classic Zermelo-Fraenkel set theory ZFC, while they are non-well-founded sets in Aczel set theory. Aczel non-well-founded set theory is an axiomatic system consisting of the ZFC axioms, except the Axiom of Foundation, which is replaced with Aczel Anti-Foundation Axiom (AFA), see also

Rathjen [4]. Situation Theory, when based on Aczel non-well-founded sets, models circular information and self-reference, including for concepts. It can model potentially large classes of situation-theoretic objects, which theoretically are proper non-well-founded sets. What is significant, for practical, intelligent applications, is that situation-theoretic objects, even when properly non-well-founded by the AFA, have finite, not necessarily large representations, e.g., visualized as cyclic graphs. Large objects can be limited for practical applications, e.g., by restrictions from specific domains of applications.

Situation Theory has been under development as a theory of the inherently relational and situational nature of information, in general, not only of linguistic meanings, by diverging from the traditional possible-world theories of semantics with type-theoretic settings, in particular from Montague's IL (see Montague [5]). Detailed discussions and motivations of the situation-theoretic objects, such as situation types similar to the ones introduced in this article, are given in Barwise and Perry [2]. For an informal introduction to Situation Theory and Situation Semantics for human language, with examples and intuitions, see Devlin [6]. Note that the typed situation-theoretic objects that we use in this paper extend the ideas of situated objects in the early works on Situation Theory, and in addition, are strictly defined objects of mathematical structures. Formal introduction, in the lines of our work here, is given in Loukanova [7]–[10]. These works include examples from human language, while Loukanova [11], [12] provides syntax-semantics constructions of human language expressions, by using phrase-structure syntax, which is the precursor and theoretic backbone of parsers in currently prevailing computational syntax.

One of the distinguished applications of Situation Theory has been Situation Semantics for semantic representations of human language in computational grammars. Head-driven Phrase Structure Grammar (HPSG) is one of the first practical grammar frameworks, based on formal syntax of human language by using typed, linguistic feature-value structures, see Pollard and Sag [13], [14], and Sag et al. [15]. Originally, HPSG was introduced by the ideas of Situation Theory for distribution of partial information throughout grammatical representations, via typed feature-value structures. Various, partly specified feature-structures can be combined according to grammar principles and constraints, by unification and expanding them with new information. From start, HPSG came with ambitions to use Situation Semantics for including

semantic representations in syntactic analyses. Current HPSG systems have been successfully realizing such semantic representations with a specialized language, Minimal Recursion Semantics (MRS), for handling scope ambiguities, see, e.g., Copestake et al. [16]. Loukanova [17] shows that the concept of minimal recursion in MRS has a functional formalization by the formal language of acyclic recursion introduced in Moschovakis [18]. By considering the relational character of the predicate symbols used in elementary predications in MRS, we see MRS as an implementation of a special case of a formal language for Situation Theory, in the lines of Loukanova [10], while more work on the relationship is necessary. The original Situation Semantics inspired other work in linguistics. E.g., it was used for semantic analysis of questions, see Ginzburg and Sag [19]. Lambalgen and Hamm [20] used concepts of Situation Semantics for semantics of tense and aspect, from cognitive perspective.

Situation Theory is an open area of theoretic development, with potentials for varieties of applications. While it has established classic applications to computational semantics, as briefly summarized above, both Situation Theory and Situation Semantics are largely open areas, in theory and applications. Currently, Situation Theory has new significance as a theory of heterogeneous information, along with the proliferation of interdisciplinary technologies and applications, especially in Artificial Intelligence and other areas that involve intelligent computation.

This paper is on a specific task of using situated information, with parametric objects, to represent hierarchically linked classes of parametric concepts. We employ situated types that support linking parametric objects with restrictions. We introduce primary restrictions over parametric objects as types associated with the argument roles of relations and types. These restrictions are called appropriateness conditions over argument roles. The argument roles (commonly known as argument slots) of relations and types can be filled up only by objects satisfying the respective appropriateness conditions.

The notational symbolism that we use to designate abstract objects of Situation Theory reminds of expressions of a formal language, but by these notations, we do not define a formal language and do not use any such formal language per se. I.e., Situation Theory is a higher-order, typed, mathematical structure. In this work, we use Situation Theory as a model theory of information, by a focus on specific abstract objects, without formal language. On the other side, a formal language for Situation Theory can provide many advantages. E.g., it is important to have a formal language for situation-theoretic analyses of human and artificial languages, via semantic representations by formal terms (which are usually called logic forms). Introducing a formal language for Situation Theory is the topic of other work, see Loukanova [10], [21], for development of formal languages of Situation Theory. In this work, our focus is on introducing semantic domains of situation-theoretic objects, i.e., complex types with parameters and complex, restricted sets of linked parameters. In the second part of the paper, we use situation-theoretic objects to represent

parametric concepts, i.e., concepts as model-theoretic objects with rich informative structure, where information can be parametric.

Note 1: The situation-theoretic objects are often designated by multi-line expressions, i.e., spread over several lines, for lack of space in a single line, but also to visualize the structure of the objects. We have tried to follow traditional indentation as in programming, wherever possible, but primarily, we try to follow the convention that the arguments of a given relation or type are vertically aligned.

II. A BRIEF OVERVIEW OF BASIC SITUATION-THEORETIC NOTIONS

This section introduces situation-theoretic notions and objects that are fundamental for representation of information and essential for the following sections of the paper. Informally, the informational pieces, called *infons*, are basic and complex objects that have structure carrying information about relations and objects filling the arguments of the relations, at time-space locations. Recursively, basic and complex infons are constructed by starting with primitive relations, argument roles, primitive individuals filling the argument roles of the relations, basic space-time locations, and positive or negative polarity. The polarity of an infon carries the information about whether or not the objects in the infon are in its relation.

A. Primitive Individuals

A collection (typically, a set) \mathcal{A}_{IND} is designated as the set of primitive *individuals* of Situation Theory:

$$\mathcal{A}_{\text{IND}} = \{a, b, c, \dots\} \quad (1)$$

The objects in \mathcal{A}_{IND} are set-theoretic objects, not necessary atomic urelements, which are considered as primitives in Situation Theory. In various versions of Situation Theory, designated for specific applications, some of the individuals in \mathcal{A}_{IND} may be parts of other individuals in \mathcal{A}_{IND} , and as such, can be in respective *part-of* relations.

B. Space-time Locations

Situation Theory make a substantial use of a given class \mathcal{A}_{LOC} of space-time points, periods, and regions units. Note that \mathcal{A}_{LOC} can be a proper class, or a set, depending on the version of Situation Theory that one can select to use in applications.

$$\mathcal{A}_{\text{LOC}} = \{l, l_0, l_1, \dots\} \quad (2)$$

The collection \mathcal{A}_{LOC} is endorsed with relations of time precedence \prec , time overlapping \circ_t , space overlapping \circ_s , space-time overlapping \circ , and space and time inclusions \subseteq_t , \subseteq_s , \subseteq . In some versions of Situation Theory, the space-tile locations can be given by complex objects, e.g., as pairs of two components, one for space locations (regions), and one for time points or periods.

C. Primitive Relations

Significantly, Situation Theory has a collection (which can be a set in practical applications, or a proper class) \mathcal{A}_{REL} of abstract, primitive objects that are the primitive, i.e., basic, relations: $\mathcal{A}_{\text{REL}} = \{r_0, r_1, \dots\}$. The elements of \mathcal{A}_{REL} are abstract representatives of real or virtual relations. For example, some of them can be abstract representatives of real properties of objects and relations between objects, in reality, or in virtual models, which humans are attuned to distinguish perceptually in reality, or cognitively, i.e., conceptually.

In typical set-theoretic practice, relations between set-theoretic objects are represented as sets of ordered tuples of the objects being in the relevant relations. On the contrary, Situation Theory takes the relations in \mathcal{A}_{REL} as primitive, first-class objects. I.e., the objects in \mathcal{A}_{REL} , are primitive objects that are not themselves sets of tuples of individuals being in those relations. Set-theoretically, the primitive relations in \mathcal{A}_{REL} , as well as the other primitive objects in Situation Theory, such as individuals, properties, relations, and types, can be taken as urelements of the meta-theoretic set theory. E.g.,

$$\mathcal{A}_{\text{REL}} = \{man, woman, dog, run, like, \dots\} \quad (3)$$

By introducing more complex situation-theoretic objects, it is possible to define the notion of the *extension* of a given relation r , in a given situation s as the set of the tuples of objects being in the relation r in s . For example, the extension of the relation *read* in a given, specific situation s and a space-time location l , is the set of all pairs $\langle a, b \rangle$ of objects, such that the primitive relation of reading holds between a as the reader and b as the object that is read, at the location l , in the situation s .

D. Primitive Types

Situation Theory has a collection (a relatively small, finite set) of objects, which are called *primitive* or *basic* types, that represent our intuitions, cognitive concepts of types, and type classifications of objects in specific areas of applications:

$$B_{\text{TYPE}} = \{\text{IND, LOC, REL, TYPE, POL, PAR,} \quad (4a)$$

$$\text{ARoles, INFON, SIT, PROP, } \models \} \quad (4b)$$

where IND is the type of individuals; LOC: of space-time locations; REL: of relations (primitive and complex); TYPE: of types (primitive and complex); POL: of two polarity objects (e.g., represented by the natural numbers 0 and 1); PAR: of parameters; ARoles: of abstract argument roles (primitive and complex); INFON: of situation-theoretic objects that are basic or complex information units; SIT: of situations; PROP: of abstract objects that are propositions; \models is a type called “supports”. Some of these types will be explained later.

E. Primitive Parameters — Indeterminates

Situation Theory has a collection (a set) of primitive parameters, for each of the basic types, e.g.:

$$\mathcal{P}_{\text{IND}} = \{\dot{a}, \dot{b}, \dot{c}, \dots\}, \quad \mathcal{P}_{\text{LOC}} = \{\dot{l}_0, \dot{l}_1, \dots\}, \quad (5a)$$

$$\mathcal{P}_{\text{REL}} = \{r_0, r_1, \dots\}, \quad \mathcal{P}_{\text{POL}} = \{\dot{i}_0, \dot{i}_1, \dots\}, \quad (5b)$$

$$\mathcal{P}_{\text{SIT}} = \{\dot{s}_0, \dot{s}_1, \dots\}, \quad \dots \quad (5c)$$

We assume that, for every type $\theta : \text{TYPE}$, there is potential availability of parameters of that type θ , (6a).

$$\mathcal{P}_\theta \text{ is a class of parameters, for every } \theta : \text{TYPE} \quad (6a)$$

$$p \in \mathcal{P}_\theta \quad \text{iff} \quad p : \theta \text{ and } p : \text{PAR} \quad (6b)$$

Thus, theoretically, the classes of types and parameters can be proper classes, which are Aczel non-well founded sets, see Aczel [3]. Note that in applications, for many types $\theta : \text{TYPE}$, it can be the case that $\mathcal{P}_\theta = \emptyset$. Practically, it would be useful, to add classes \mathcal{P}_θ not in advance, but depending on needs, and to add fresh, new parameters to them “on-the-go”.

Sometimes, but not always, we use a notation originally introduced in Situation Theory, to denote parameters with dotted letters, as in (5a). Marking letters with dots is a visual distinction of parameters from other individuals and objects. However, we should stress that this paper is about modeling domains of Situation Theory, not about a formal language. Letters, characters, and expressions that we use are notational means of denoting objects in situational domains.

In this paper, we start with the idea of situation-theoretic parameters as representing very primitive concepts that are distinguished only by their types. Thus, \mathcal{P}_{IND} is the class of the primitive concepts of individuals, e.g., \dot{a} , \dot{b} , etc., are concepts of individuals. \mathcal{P}_{LOC} is the class of the primitive concepts of space-time locations, e.g., \dot{l}_0 , \dot{l}_1 , etc., are concepts of space-time locations. \mathcal{P}_{REL} is the class of the primitive concepts of relations, where any element $r \in \mathcal{P}_{\text{REL}}$ is a concept of a relation. E.e., *blue*, as a unary relation, i.e., a property of objects, is the concept of an object being blue in color, in space-time. *give* is the concept of a relation between three objects, one being an individual giving an object to another individual, which takes place in space-time. Typically, relations between objects and properties of objects happen in space-time. The class \mathcal{P}_{SIT} consists of primitive parameters that represent abstract concepts of situations. We consider that the biological nervous systems, at least those of humans and other advanced living organisms, are attuned to recognize both abstract entities and specific instances of abstract entities. In particular, human brain has inner facilities to form and comprehend concepts for individuals, relations, space-time locations, and situations, as well as specific representatives, i.e., instantiations, of the abstract concepts.

Notation 1: For any given type T (primitive or complex) and an object θ , we write $(T : \theta)$ to designate the proposition that θ is of type T , and $T : \theta$ iff θ is of type T . An alternative notation of can be used, i.e., $\theta : T$, as in some type systems, such as the intensional logics of Montague and Gallin. We allow both notations depending on convenience, i.e., given a type T (primitive or complex) and an object θ , we write

$$(T : \theta) \quad \text{iff} \quad (\theta : T) \quad (7a)$$

i.e., the proposition that θ is of type T

$$T : \theta \quad \text{iff} \quad \theta : T \quad \text{iff} \quad \theta \text{ is of type } T \quad (7b)$$

The alternative notations in (7a)–(7b) can be used depending on the context, which makes clear the usage. Note that $(T : \theta)$

and $(\Theta : T)$ in (7a) both designate the proposition that Θ is of type T , while the alternatives in (7b) designate the verified proposition, when factually Θ is of type T .

F. Primitive Argument Roles

We assume a collection (a set) of primitive objects $\mathcal{BA}_{\text{ARoles}}$ designated as *primitive argument roles*, which is a sub-collection of the class of *complex argument roles*:

$$\mathcal{BA}_{\text{ARoles}} = \{\rho_0, \dots, \rho_n, \dots\} \subset \mathcal{A}_{\text{ARoles}} \quad (8)$$

A set of argument roles is associated with each of the primitive relations, and each of the primitive types, by a function ARGGR, with domain and range: $\text{Dom}(\text{ARGGR}) = \mathcal{A}_{\text{REL}} \cup \mathcal{B}_{\text{TYPE}}$, and $\text{Range}(\text{ARGGR}) \subseteq \text{TYPE} \times \mathcal{A}_{\text{ARoles}}$. Thus the argument roles of each type and each relation X (basic or complex, recursively for the complex ones) are associated with corresponding types that restrict what objects can fill up the argument roles. I.e., every relation or type X , is associated with argument roles:

$$\text{ARGGR}(X) = \{T_1 : \text{arg}_1, \dots, T_n : \text{arg}_n\}, \quad (9a)$$

$$\text{where } \text{arg}_i : \text{ARoles} \text{ and } T_i : \text{TYPE}, i = 1, \dots, n, \quad (9b)$$

for some $n \geq 0$

The types T_i are called *appropriateness constraints* of the corresponding argument roles arg_i , $i = 1, \dots, n$, of the relation (type) X . Complex relations and types are associated with argument roles and corresponding appropriateness constraints, according to recursive definitions given in what follows, supplemented by examples.

In what follows, we assume that if an argument role arg_i of a relation or type X , (9) is restricted by a type $T_i : \text{arg}_i$, this argument role can be filled up by a situation-theoretic object of type T_i , including by parameters, e.g., by using (6b).

G. Basic Infons

Basic infons can be represented by specialized, marked tuples

$$\langle \text{infor}, \gamma, \theta, \tau, i \rangle \quad (10)$$

where $\gamma \in \mathcal{R}_{\text{REL}}$ is a relation (primitive or complex), $\text{LOC} : \tau$, $\text{POL} : i$, and θ is a function, called the *argument-role filling* of γ , which fills up the argument roles $\text{arg}_1, \dots, \text{arg}_n$ ($n \geq 0$) of γ with objects ξ_1, \dots, ξ_n of respective types T_1, \dots, T_n , i.e.:

$$\theta = \{\langle T_1 : \text{arg}_1, \xi_1 \rangle, \dots, \langle T_n : \text{arg}_n, \xi_n \rangle\} \quad (11)$$

for some situation-theoretic objects ξ_1, \dots, ξ_n satisfying the corresponding appropriateness constraints of the argument roles of the relation γ .

Notation 2: The basic infons (10), as well as some of the complex ones are denoted by (12):

$$\ll \gamma, \theta, \tau, i \gg \quad (12)$$

Notation 3: When the types of the argument roles are agreed, i.e., understood by the context, the function filling the argument roles is denoted by (13).

$$\theta = \{\langle \text{arg}_1, \xi_1 \rangle, \dots, \langle \text{arg}_n, \xi_n \rangle\} \quad (13)$$

H. Complex Infons

Complex infons for representation of conjunctive and disjunctive information are formed by the operators conjunction and disjunction. In some earlier versions of Situation Theory, the operators conjunction and disjunction in the infon constructions were taken to be primitive relations between infons, for which locations are irrelevant.

Notation 4: For sake of space, by assuming that the order of the argument roles is agreed to avoid confusion, we shall adopt the traditional linear notations of the basic infons, with or without the type constraints over the argument roles, as in (14a)–(14d).

$$\ll \gamma, T_1 : \text{arg}_1 : \xi_1, \dots, T_n : \text{arg}_n : \xi_n, \quad (14a)$$

$$\text{LOC} : \tau; i \gg$$

$$\ll \gamma, \text{arg}_1 : \xi_1, \dots, \text{arg}_n : \xi_n, \text{LOC} : \tau; i \gg \quad (14b)$$

$$\ll \gamma, T_1 : \xi_1, \dots, T_n : \xi_n, \tau; i \gg \quad (14c)$$

$$\ll \gamma, \xi_1, \dots, \xi_n, \tau; i \gg \quad (14d)$$

We denote the class of all infons, basic or complex, by $\mathcal{I}_{\text{INFON}}$.

III. BASIC SEMANTIC CONCEPTS

In this section, we introduce the idea of representing basic semantic concepts as situation-theoretic objects with parametric components. We use prototypical examples of semantic relations and parametric information peaces.

Example 3.1: Assume that $\text{read}_2 \in \mathcal{A}_{\text{REL}}$, i.e., read_2 is a primitive situation-theoretic object of type REL, i.e., by (7b), $\text{read}_2 : \text{REL}$. Assume also that the relation read_2 has two argument roles as in (15):

$$\text{ARGGR}(\text{read}_2) = \{\text{IND} : \text{reader}, \text{IND} : \text{readed}\} \quad (15)$$

for $\text{reader}, \text{readed} \in \mathcal{A}_{\text{ARoles}}$

Then, we have the following infons in (16a)–(16d).

$$\ll \text{read}_2, \text{IND} : \text{reader} : a, \quad (16a)$$

$$\text{IND} : \text{readed} : b, \text{LOC} : l; 1 \gg$$

$$\ll \text{read}_2, \text{IND} : \text{reader} : \dot{a}, \quad (16b)$$

$$\text{IND} : \text{readed} : \dot{b}, \text{LOC} : \dot{l}; 1 \gg$$

$$\ll \text{read}_2, \text{IND} : \text{reader} : a, \quad (16c)$$

$$\text{IND} : \text{readed} : \dot{b}, \text{LOC} : \dot{l}; 1 \gg$$

$$\ll \text{read}_2, a, b, l; \dot{p} \gg \quad (16d)$$

Note that we use the “misspelled” notation *readed* for the semantic argument role of the relation read_2 , which is to be filled by the object that is being read. I.e., $\text{readed} \in \mathcal{A}_{\text{ARoles}}$ is an abstract argument role denoted by this “misspelled” variant of the past participle of the verb “read”. This notation is by a trend in the early versions of Situation Semantics, by which, the argument role for the actor of an activity, usually denoted by a verb, is represented by using the suffix “er”, and the argument role for the object acted upon by using the suffix “ed” or “en”. Thus, some argument roles can have “misspelled” notations. How the argument roles are denoted is a matter of agreement settings. We have chosen here that trend,

to avoid indexing the argument roles with natural numbers, i.e., arg_1, \dots, arg_n , which carries connotations that the argument roles have been linearly ordered, which is not always the case. We would like to stress that, in general, there is no intrinsic order over the argument roles of relations and types, except in specific cases and for notational needs.

In (16a)–(16d), $a, b \in \mathcal{A}_{IND}$ are individuals, $l \in \mathcal{A}_{LOC}$ is a location, while $\hat{a}, \hat{b} \in \mathcal{P}_{IND}$ are parameters for individuals, $\hat{l} \in \mathcal{P}_{LOC}$ is a location parameter, and $\hat{p} \in \mathcal{P}_{POL}$ is a parameter for either of the two polarities $\{0, 1\}$. E.g., the unknown individual, which fills up the argument role of the material that is being read, is represented by a semantic parameter \hat{b} that is restricted to be of type IND, by the constraint over the argument role that \hat{b} fills up, i.e., $IND : \textit{readed} : \hat{b}$. This constraint allows \hat{b} to be of both types IND and PAR, i.e., $IND : \hat{b}$ and $PAR : \hat{b}$, by using (6b). Similarly, for the constraints over $\hat{a}, \hat{l}, \hat{p}$.

We stress that the parameters are not variables in a formal language. In Situation Theory, parameters are first-class model-theoretic objects. In this paper, we use the parameters \hat{a}, \hat{b} as representing the abstract concept of individuals that are ‘unknown’, and \hat{l} represents a concept of a space-time location, without being any specifically determined location. The two specific individuals a, b , the location l , and the ‘confirming’ polarity 1, are instantiations of the corresponding concepts of two individuals, a location, and a polarity represented by parameters.

Next, we give examples for other parametric infons, by using a relation of reading, which is an alternative to the relation $read_2$ in (16a)–(16d), for having an extra argument role for an intended listener, which could have been denoted by *listener*. To avoid the connotation that the object filling up this argument role listens (which might not be the case), we denote it by *readee*.

Example 3.2: Now, we assume that $read \in \mathcal{A}_{REL}$, i.e., $read$ is a primitive situation-theoretic object of type REL, i.e., by (7b), $read : REL$. The significant difference is that, unlike in Example 3.1, here the relation $read$ has the following three argument roles.

$$ARGR(read) = \{IND : \textit{reader}, IND : \textit{readed}, \\ IND : \textit{readee}\} \quad (17)$$

for $\textit{reader}, \textit{readed}, \textit{readee} \in \mathcal{A}_{Roles}$

Now we consider the infon (18a)–(18e).

$$\ll read, IND : \textit{reader} : a, \quad (18a)$$

$$IND : \textit{readed} : \hat{b}, \quad (18b)$$

$$IND : \textit{readee} : \hat{c}, \quad (18c)$$

$$LOC : \textit{Loc} : \hat{l}; \quad (18d)$$

$$POL : \textit{Pol} : 1 \gg \quad (18e)$$

The infon (18a)–(18c) represents the information that an individual a reads the unknown or undetermined material \hat{b} (i.e., \hat{b} is a semantic parameter) to the unknown or undetermined \hat{c} (i.e., \hat{c} is a semantic parameter), at the unknown

or undetermined location \hat{l} (i.e., \hat{l} is a location parameter). The informational piece (18a)–(18e) is about the relation $read$ between specific objects a, \hat{b}, \hat{c} , taking place at the specific location \hat{l} . The difference is that $a \in \mathcal{A}_{IND}$ is explicitly given as known, determinedly picked up from the set \mathcal{A}_{IND} . While $\hat{b} \in \mathcal{P}_{IND}$, $\hat{c} \in \mathcal{P}_{IND}$, and $\hat{l} \in \mathcal{P}_{LOC}$, are also specific, but are either unknown or simply left indeterminate, i.e., as parameters.

Example 3.3:

$$\ll read, T_a : \textit{reader} : a, \quad (19a)$$

$$T_b : \textit{readed} : \hat{b}, \quad (19b)$$

$$T_c : \textit{readee} : \hat{c}, \quad (19c)$$

$$LOC : \textit{Loc} : \hat{l}; \quad (19d)$$

$$POL : \textit{Pol} : \hat{p} \gg \quad (19e)$$

In (19a)–(19e), by T_a, T_b and T_c , we represent sets of types that constrain the argument roles *reader, readed, readee* of the relation $read$, by undetermined types.

By using a parameter for polarity, the infon (19a)–(19e) represents the parametric information that the specific object a either reads or does not read the undetermined \hat{b} to the undetermined \hat{c} , at the undetermined location \hat{l} . The undetermined polarity is represented by a semantic parameter \hat{p} that is restricted to be of type POL, by the constraint over the argument role that \hat{p} fills up, i.e., $POL : \textit{Pol} : \hat{p}$. This constraint allows \hat{p} to be of both types POL and PAR, i.e., $POL : \hat{p}$ and $PAR : \hat{p}$, by using (6b). Similarly, for the constraint over \hat{l} .

The importance of using the polarity parameter \hat{p} , in this example, is that we do not have an explicit disjunction — we still have a piece of information by (19a)–(19e) about the relation of reading concerning a as a possible reader.

The reason for which we have chosen examples with the relation of reading is not only to demonstrate the definitions, but also because it is denoted by a verb that syntactically can have either one, two, or three syntactic arguments. With the Examples 3.1–3.3, we make a point for a distinction between syntactic arguments of lexemes, in this case of the verb “read”, and the corresponding semantic argument roles of the semantic relations denoted by those lexemes. Typically, a sentence like (20b) can be rendered into a term having a component infon similar to one of the infons in (16a)–(16d), which may be combined with additional infons depending on the noun phrases A and B. Thus, the verb “read”, co-occurring with two syntactic arguments, would be treated as denoting a relation $read_2$ associated with two semantic argument roles (15). Similarly, the verb “read” in a sentence like (20a) can be rendered into a relation $read_1$ associated with one semantic argument role.

$$A \textit{ reads}. \quad (20a)$$

$$A \textit{ reads B}. \quad (20b)$$

$$A \textit{ reads B to C}. \quad (20c)$$

A sentence like (20c) can be rendered into a term with a component infon similar to the one in (18a)–(18e). In this

way, when the verb “read” co-occurs with three syntactic arguments, it is rendered into a relation *read*, which is different from *read*₁ and *read*₂, by having three semantic argument roles (17). As a choice, one may keep up with this line of introducing different, variant relations, depending on the number of argument roles. This choice is deficient in representing that there is a common semantic relation of reading that may exhibit only some of its semantic argument roles in language expressions like (20a)–(20c).

Here we point that by using semantic parameters, we can render the verbal lexeme “read”, occurring in the three kinds of sentences like (20a)–(20c), into the same relation *read*, associated with three semantic argument roles (17). For sentences like (20a), *read* would be rendered into the relation *read*, by filling up each of the argument roles *readed* and *readee* with undetermined parameters. The role *reader* can be filled up by a specific individual *a*, or by a parameter *á* along with additional information depending on the NP A. In sentences like (20b), *read* would be rendered into the same relation *read*, by filling up the argument role *readee* with an undetermined parameter, while the roles *reader* and *readed* can be filled by specific individuals, or by parameters along with additional information depending on NPs A and B. In this way, we have the same semantic relation *read*, which may exhibit only some of the semantic information associated with it, explicitly in syntactic expressions like (20a)–(20c). Information that is not expressed in (20a)–(20c) and is not available by context, is kept parametric and underspecified.

Thus, by the relation *read* associated with semantic argument roles (17), we model a general, semantic concept of reading. Its argument roles can be filled up by parameters or specific individuals. Furthermore, the parameters can be additionally specified as we show in the second part of the paper. We used *read* as a prototypical example of a class of similar basic semantic concepts.

IV. PROPOSITIONS

Definition 1 (Proposition): A proposition is a semantic, situation-theoretic object, represented set-theoretically in Aczel non-well-founded sets, by the tuple $\langle \text{PROP}, \mathbb{T}, \theta \rangle$, where $\mathbb{T} \in \mathcal{T}_{\text{TYPE}}$ is a type that is associated with a set of argument roles (21)

$$\text{ARGR}(\mathbb{T}) = \{T_1 : \text{arg}_1, \dots, T_n : \text{arg}_n\} \quad (21)$$

and θ is a function, called the *argument-role filling* of the type \mathbb{T} , which fills up the argument roles $\text{arg}_1, \dots, \text{arg}_n$ ($n \geq 0$), of \mathbb{T} with objects ξ_1, \dots, ξ_n of respective types T_1, \dots, T_n , i.e.:

$$\theta = \{\langle T_1 : \text{arg}_1, \xi_1 \rangle, \dots, \langle T_n : \text{arg}_n, \xi_n \rangle\} \quad (22)$$

for some situation-theoretic objects ξ_1, \dots, ξ_n satisfying the corresponding appropriateness constraints of the argument roles of the type \mathbb{T} , i.e.:

$$T_1 : \xi_1, \dots, T_n : \xi_n \quad (23)$$

Notation 5: Typical notations are

$$\langle \text{PROP}, \mathbb{T}, \theta \rangle \equiv \langle \mathbb{T}, \theta \rangle \quad (24a)$$

$$\equiv (\mathbb{T} : \theta) \quad (24b)$$

$$\equiv (\theta : \mathbb{T}) \quad (24c)$$

The notations (24a) and (24b) resemble the application operators, where “ \mathbb{T} applies to the argument(s) θ ”. The notational variants (24b) and (24c) are used alternatively depending on the context and the specific types \mathbb{T} . The notation (24c) follows the verbal expression “the proposition that the object(s) filling up the argument role(s) of \mathbb{T} are of type \mathbb{T} ”.

Definition 2 (Situated propositions): Situated proposition, instantiated in a situation s , is any proposition (25):

$$\langle \text{PROP}, \models, \text{SIT} : s, \text{INFON} : \sigma \rangle, \quad (25)$$

where s is a situation parameter $s \in \mathcal{P}_{\text{SIT}}$ i.e., $\text{SIT} : s$ and σ is a basic or complex infon, i.e., $\sigma \in \mathcal{I}_{\text{INFON}}$.

The type \models (pronounced “supports”) has two argument roles, (26):

$$\text{ArgR}(\models) = \{\text{SIT} : \text{arg}_{\text{SIT}}, \text{INFON} : \text{arg}_{\text{INFON}}\} \quad (26)$$

A proposition (25) is pronounced “the proposition that the situation s supports the infon σ ”, or “the proposition that σ holds in the situation s ”.

Notation 6:

$$\langle \text{PROP}, \models, \text{SIT} : s, \text{INFON} : \sigma \rangle \equiv \langle \text{PROP}, \models, s, \sigma \rangle \quad (27a)$$

$$\equiv \langle \models, s, \sigma \rangle \quad (27b)$$

$$\equiv (s \models \sigma) \quad (27c)$$

V. COMPLEX TYPES AND RELATIONS

Situation Theory has an abstraction operator, which resembles the λ -abstraction in functional λ -calculi, but is model-theoretic, informational abstraction. The informational abstraction is not a syntactic construction of a λ -expression in a formal language. It defines abstract complex relations and complex types, with abstract argument roles. These abstract situation-theoretic objects can be modeled with set-theoretic objects, by choosing appropriate set theory, e.g., a classic set-theory for more restricted applications, while choosing Aczel non-well-founded set theory, such as in Aczel [3], for more sophisticated applications.

A. Complex Relations

Definition 3 (Complex relations and argument roles): Let σ be a given infon, and $\{\xi_1, \dots, \xi_n\}$ a set of parameters, i.e., primitive or complex objects of type PAR , $\text{PAR} : \xi_i$, for $i = 1, \dots, n$ ($n \geq 0$), which may occur in σ (when some ξ_i does not occur in σ , the abstraction over ξ_i is vacuous, but it adds an additional argument role to the complex relation). Let, for each $i \in \{1, \dots, n\}$, T_i be the union of the constraints over the argument roles filled up by ξ_i . Then $\lambda\{\xi_1, \dots, \xi_n\}\sigma$ is a *complex relation*, with abstract argument roles denoted by

$[\xi_1], \dots, [\xi_n]$ and having T_1, \dots, T_n as *appropriateness (type) constraints*, respectively, i.e.:

$$[T_1 : [\xi_1], \dots, T_n : [\xi_n] \mid \sigma] \in \mathcal{T}_{\text{REL}}, \quad \text{and} \quad (28a)$$

$$\begin{aligned} \text{ARGR}([T_1 : [\xi_1], \dots, T_n : [\xi_n] \mid \sigma]) \\ = \{ \langle [\xi_1], T_1 \rangle, \dots, \langle [\xi_n], T_n \rangle \} \\ \equiv \{ T_1 : [\xi_1], \dots, T_n : [\xi_n] \} \end{aligned} \quad (28b)$$

Instead of (28a), we shall primarily use the notation (29b), by suppressing the types of the argument roles when they are understood. The notation (29a) may be useful too.

$$[T_1 : [\xi_1], \dots, T_n : [\xi_n] \mid \sigma] \quad (29a)$$

$$\equiv \{ \lambda(\xi_1), \dots, \lambda(\xi_n) \} \sigma \quad (29b)$$

$$\equiv \lambda\{ \xi_1, \dots, \xi_n \} \sigma \quad (29c)$$

Upon agreed order, which is a typical practice in mathematics and in computer science, the argument roles (usually called argument slots) and/or the types constraints can be skipped, and only the objects filling up the argument roles (slots) are listed.

B. Complex Types

In this subsection, we define the abstract objects complex types. They are significant for what follows in this paper.

Definition 4 (Complex types and argument roles): Assume that

- 1) Θ is a given situation-theoretic proposition, and ξ_1, \dots, ξ_n are parameters, i.e., $\text{PAR} : \xi_i$, for $i = 1, \dots, n$ ($n \geq 0$).
- 2) For each $i \in \{1, \dots, n\}$, T_i is the union of all the appropriateness constraints of all the argument roles of constituents of Θ that are filled up by ξ_i .

Note that Θ may have various, components (constituents), which are either types or relations, with arguments roles that are filled by ξ_i . In addition, a single constituent type, or a constituent relation, may have more than one argument role filled by ξ_i .

The result of the *abstraction over the parameters* ξ_1, \dots, ξ_n from the proposition Θ is a *complex type* (30a), with argument roles $[\xi_i]$ that are associated with *appropriateness (type) constraints*, respectively, $(T_i : [\xi_i])$, for $i = 1, \dots, n$ ($n \geq 0$), i.e., (30b):

$$[T_1 : [\xi_1], \dots, T_n : [\xi_n] \mid \Theta] \in \mathcal{T}_{\text{TYPE}}, \quad \text{and} \quad (30a)$$

$$\begin{aligned} \text{ARGR}([T_1 : [\xi_1], \dots, T_n : [\xi_n] \mid \Theta]) \\ = \{ \langle [\xi_1], T_1 \rangle, \dots, \langle [\xi_n], T_n \rangle \} \\ = \{ T_1 : [\xi_1], \dots, T_n : [\xi_n] \} \end{aligned} \quad (30b)$$

Thus the abstraction over the parameters ξ_1, \dots, ξ_n , from the proposition Θ , results in complex, abstract argument roles $[\xi_1], \dots, [\xi_n]$ of the complex type $[T_1 : [\xi_1], \dots, T_n : [\xi_n] \mid \Theta]$. The abstraction creates argument roles along with appropriateness type constraints $T_1 : [\xi_1], \dots, T_n : [\xi_n]$.

Note that, as in the complex relations, the parameters ξ_i , $i = 1, \dots, n$, are primitive or complex objects of type PAR , $\text{PAR} : \xi_i$, for $i = 1, \dots, n$ ($n \geq 0$), which may occur in Θ .

When some ξ_i does not occur in Θ , the abstraction over $T_i : \xi_i$ is vacuous, but it adds an argument role of the complex type (30a). In the cases when $n = 0$, the complex type has no argument roles.

Notation 7: A complex type $[T_1 : [\xi_1], \dots, T_n : [\xi_n] \mid \Theta]$ is alternatively denoted by (31b), when the argument roles of the result of the abstraction operation are suppressed; and with (31c) when the corresponding type constraints over the complex roles are suppressed too.

$$[T_1 : [\xi_1], \dots, T_n : [\xi_n] \mid \Theta] \quad (31a)$$

$$\equiv [T_1 : \xi_1, \dots, T_n : \xi_n \mid \Theta] \quad (31b)$$

$$\equiv [\xi_1, \dots, \xi_n \mid \Theta] \quad (31c)$$

$$\equiv \lambda\{ \xi_1, \dots, \xi_n \} \Theta \quad (31d)$$

Similarly to the complex relations in (29a)–(29c), complex types might sometimes be denoted by the λ -notation $\lambda\{ \xi_1, \dots, \xi_n \} \Theta$ in (31d). Whether a situation-theoretic object $[T_1 : \xi_1, \dots, T_n : \xi_n \mid \vartheta]$, as an abstraction over the parameters $\text{PAR} : \xi_1, \dots, \text{PAR} : \xi_n$ in an object ϑ , is a complex relation or complex type depends on whether ϑ is an infon or a proposition, not on what notation we use for it per se. Nevertheless, we shall primarily use the notations in (29b)–(29c) for relations, and (31a)–(31c) for types, in order to make clear distinction between (1) the abstract complex relations, which are abstractions over parameters from infons; and (2) the abstract complex types, which are abstractions over parameters from propositions.

VI. COMPLEX CONCEPTS

In this section, we demonstrate how to use complex parametric types to model taxonomic concepts. The concepts can be at different level of abstraction over various parameters.

A. A Sample Concept

Here we demonstrate the general ideas of concepts in taxonomic classifications, e.g., the class of *odd-toed, ungulate* entities as a subclass of *animate* entities. Then, we demonstrate how to instantiate the concept of an odd-toed, ungulate entity with a specific representative a having a complex property, e.g., walking in a space-time location and sleeping in another space-time location.

$$T_{\text{animate}} \equiv \quad (32a)$$

$$[\text{IND} : [x] \mid (\dot{s} \models \ll \text{animate}, \text{IND} : \text{arg} : x, \text{LOC} : \dot{l}; 1 \gg)] \quad (32b)$$

The type T_{otu} defined in (33a)–(33c) is the type of odd-toed ungulate individuals in some (parametric) situation \dot{s} and location \dot{l} :

$$T_{\text{otu}} \equiv [T_{\text{animate}} : [x] \mid (\dot{s} \models \quad (33a)$$

$$\ll \text{odd-toed}, T_{\text{animate}} : \text{arg} : x, \text{LOC} : \dot{l}; 1 \gg \wedge \quad (33b)$$

$$\ll \text{ungulate}, T_{\text{animate}} : \text{arg} : x, \text{LOC} : \dot{l}; 1 \gg)] \quad (33c)$$

B. Instantiation of Concepts

The type (34a) is a two-argument type of a situation and a location, in which a specific odd-toed ungulate individual a walks. Note that the type (34a) is defined only in case the proposition ($a : T_{otu}$) is true, i.e., $a : T_{otu}$, which is so in case a of type T_{otu} .

$$[\text{SIT} : [\dot{s}_0], \text{LOC} : [\dot{l}_0] \mid (\dot{s}_0 \models \quad (34a)$$

$$\ll \text{walk}, T_{otu} : \text{walker} : a, \text{LOC} : \dot{l}_0; 1 \gg)] \quad (34b)$$

The type (35a)-(35d) is a three-argument type of situations \dot{s}_0 and locations \dot{l}_0, \dot{l}_1 , such that, in the situation \dot{s}_0 , the odd-toed ungulate individual a walks through the location \dot{l}_0 , and sleeps through the location \dot{l}_1 , and where the locations \dot{l}_0 and \dot{l}_1 are not overlapping in time.

$$[\text{SIT} : [\dot{s}_0], \text{LOC} : [\dot{l}_0], \text{LOC} : [\dot{l}_1] \mid \quad (35a)$$

$$(\dot{s}_0 \models \ll \text{walk}, T_{otu} : \text{walker} : a, \text{LOC} : \dot{l}_0; 1 \gg) \wedge \quad (35b)$$

$$\ll \text{sleep}, T_{otu} : \text{sleeper} : a, \text{LOC} : \dot{l}_1; 1 \gg) \wedge \quad (35c)$$

$$(\dot{l}_0 \not\propto_t \dot{l}_1)] \quad (35d)$$

The type (36a)-(36d) is a four-argument type of situations \dot{s}_0, \dot{s}_1 and locations \dot{l}_0, \dot{l}_1 , such that, in the situation \dot{s}_0 , the odd-toed ungulate individual a walks through the location \dot{l}_0 , and, in the situation \dot{s}_1 , a sleeps through the location \dot{l}_1 , where the locations \dot{l}_0 and \dot{l}_1 are not time overlapping.

$$[\text{SIT} : [\dot{s}_0], \text{LOC} : [\dot{l}_0], \text{SIT} : [\dot{s}_1], \text{LOC} : [\dot{l}_1] \mid \quad (36a)$$

$$(\dot{s}_0 \models \ll \text{walk}, T_{otu} : \text{walker} : a, \text{LOC} : \dot{l}_0; 1 \gg) \wedge \quad (36b)$$

$$(\dot{s}_1 \models \ll \text{sleep}, T_{otu} : \text{sleeper} : a, \text{LOC} : \dot{l}_1; 1 \gg) \wedge \quad (36c)$$

$$(\dot{l}_0 \not\propto_t \dot{l}_1)] \quad (36d)$$

C. Concept Subdivision

A conceptual class can be subdivided into subclasses by using subtypes. E.g., we demonstrate the technique, by the type $T_{nocturnal}$ in (37a) defined as a subtype of the type T_{otu} defined in (33a)-(33c).

$$T_{nocturnal} \equiv [\text{IND} : [x] \mid \quad (37a)$$

$$(\dot{s} \models \ll \text{nocturnal}, \text{IND} : x, \text{LOC} : \dot{l}; 1 \gg)] \quad (37b)$$

Now, the type $T_{nocturnal-otu}$ in (38a)-(38f), has a single argument role $[\dot{a}]$:

$$T_{nocturnal-otu} = [\{T_{nocturnal}, T_{otu}\} : \dot{a} \mid \quad (38a)$$

$$(\dot{s}_2 \models \ll \text{healthy}, T_{nocturnal} : \text{arg} : \dot{a}, \quad (38b)$$

$$\text{LOC} : \dot{l}_2; 1 \gg) \wedge \quad (38c)$$

$$(\dot{s}_0 \models \ll \text{walk}, T_{otu} : \text{walker} : \dot{a}, \text{LOC} : \dot{l}_0; 1 \gg) \wedge \quad (38d)$$

$$(\dot{s}_1 \models \ll \text{sleep}, T_{otu} : \text{sleeper} : \dot{a}, \text{LOC} : \dot{l}_1; 1 \gg) \wedge \quad (38e)$$

$$(\dot{l}_0 \not\propto_t \dot{l}_1) \wedge (\dot{l}_0 \subset \dot{l}_2) \wedge (\dot{l}_1 \subset \dot{l}_2)] \quad (38f)$$

The type $T_{nocturnal-otu}$ in (38a)-(38f), that has a single argument role $[\dot{a}]$ with the argument constraint $\{T_{nocturnal}, T_{otu}\}$ for appropriate filling, i.e.:

$$\text{ARGR}(T_{nocturnal-otu}) = \{ \{T_{nocturnal}, T_{otu}\} : [\dot{a}] \} \quad (39)$$

Now, while the types in Examples VI-A-VI-C share the parameters for a situation s and a location l , we might consider them as related by them, as long as these types are used together, as if in a “package”. In case these types are “separated”, the parameters s and l can be anchored, i.e., instantiated, with unrelated objects of the respective types.

D. Instances of Concepts in Conceptual Sub-divisions

Let c_1 be a parameter assignment for the type $T_{nocturnal-otu}$ in (38a)-(38f), such that:

$$c_1(\dot{s}_0) = s_0, \quad c_1(\dot{l}_0) = l_0, \quad (40a)$$

$$c_1(\dot{s}_1) = s_1, \quad c_1(\dot{l}_1) = l_1, \quad (40b)$$

$$c_1(\dot{s}_2) = s_2, \quad c_1(\dot{l}_2) = l_2. \quad (40c)$$

Then the following propositions have the same truth values:

$$(c_1(T_{nocturnal-otu}) : a) \text{ is true} \quad (41a)$$

$$\iff$$

$$(c_1(T_{nocturnal}) : a) \wedge \quad (41b)$$

$$(c_1(T_{otu}) : a) \wedge \quad (41c)$$

$$(s_2 \models \ll \text{healthy}, T_{nocturnal} : \text{arg} : a, \quad (41d)$$

$$\text{LOC} : l_2; 1 \gg) \wedge$$

$$(s_0 \models \ll \text{walk}, T_{otu} : \text{walker} : a, \quad (41e)$$

$$\text{LOC} : l_0; 1 \gg) \wedge$$

$$(s_1 \models \ll \text{sleep}, T_{otu} : \text{sleeper} : a, \quad (41f)$$

$$\text{LOC} : l_1; 1 \gg) \wedge$$

$$(l_0 \not\propto_t l_1) \wedge (l_0 \subset l_2) \wedge (l_1 \subset l_2) \quad (41g)$$

In general, in order to model parametric assignments and cognitive concepts, including for semantics of natural language, and in systems of taxonomic information, we allow the sub-propositions $(c_1(T_{nocturnal}) : a)$ and $(c_1(T_{otu}) : a)$ may be true by “preserving” parametric information via assignment $c_1(\dot{s}) = \dot{s}$ and $c_1(\dot{l}) = \dot{l}$. Then, the following propositions are truth equivalent:

$$(c_1(T_{nocturnal}) : a) \iff (T_{nocturnal} : a) \quad (42a)$$

$$\iff (c_2(T_{nocturnal}) : a) \quad (42b)$$

In the above types, there is not explicit requirement that further update of information, by a new parameter assignment c_2 , have to agree on the parameters \dot{s} and \dot{l} (which in certain cases may be desirable). Let c_2 be a parameter assignment for the type $T_{nocturnal}$ in (37a), such that:

$$c_2(\dot{s}) = s', \quad c_2(\dot{l}) = l'. \quad (43)$$

Then:

$$(c_2(T_{nocturnal}) : a) \text{ is true} \quad (44a)$$

$$\iff (s' \models \ll \text{nocturnal}, \text{IND} : \text{arg} : a, \quad (44b)$$

$$\text{LOC} : l'; 1 \gg) \text{ is true}$$

$$\iff s' \models \ll \text{nocturnal}, \text{IND} : \text{arg} : a, \quad (44c)$$

$$\text{LOC} : l'; 1 \gg$$

The parameter assignment c_1 may not agree with c_2 , for example, in case $s \neq s'$ and $s \neq s'$. There is a need for explicitly expressing that c_1 should be a parameter assignment for the types $T_{nocturnal}$ in (37a), T_{otu} in (33a)–(33c), and $T_{animate}$ in (32b), so that they agree on \dot{s} and \dot{l} , as in (45):

$$c_1(\dot{s}) = s, \quad c_1(\dot{l}) = l, \quad (45)$$

The effect of (41a) is represented in (46a)–(46h):

$$(c_1(T_{nocturnal-otu}) : a) \text{ is true} \quad (46a)$$

\iff

$$(s \models \ll \text{nocturnal}, \text{IND} : \text{arg} : a, \text{LOC} : l; 1 \gg) \wedge \quad (46b)$$

$$(s \models \ll \text{odd-toed}, T_{animate} : \text{arg} : a, \text{LOC} : l; 1 \gg \wedge \quad (46c)$$

$$\ll \text{ungulate}, T_{animate} : \text{arg} : a, \text{LOC} : l; 1 \gg) \wedge$$

$$(s \models \ll \text{animate}, \text{IND} : \text{arg} : a, \text{LOC} : l; 1 \gg) \wedge \quad (46d)$$

$$(s_2 \models \ll \text{healthy}, T_{nocturnal} : \text{arg} : a, \text{LOC} : l_2; 1 \gg) \wedge \quad (46e)$$

$$(s_0 \models \ll \text{walk}, T_{otu} : \text{walker} : a, \text{LOC} : l_0; 1 \gg) \wedge \quad (46f)$$

$$(s_1 \models \ll \text{sleep}, T_{otu} : \text{sleeper} : a, \text{LOC} : l_1; 1 \gg) \wedge \quad (46g)$$

$$(l_0 \not\subset l_1) \wedge (l_0 \subset l_2) \wedge (l_1 \subset l_2) \quad (46h)$$

VII. LINKING PARAMETERS IN INFORMATIONAL STRUCTURES

The following examples are patterns of how to achieve effects of linking argument roles of relations and types and their filling. The results are complex informational structures, as in (41a)–(46h), that represent general, informational patterns.

A. Complex, Interrelated Types

The type $T_{sitatan}$ in (47a)–(47b) is a three-argument type, with argument roles for an individual, $\text{IND} : [\xi]$, a situation, $\text{SIT} : [\dot{s}']$, and a location, $\text{LOC} : [\dot{l}']$.

$$T_{sitatan} = [\text{IND} : [\xi], \text{SIT} : [\dot{s}'], \text{LOC} : [\dot{l}']] \mid \quad (47a)$$

$$(\dot{s}' \models \ll \text{animate}, \text{IND} : \text{arg} : \xi, \quad (47b)$$

$$\text{LOC} : \dot{l}'; 1 \gg)]$$

Thus, the type $T_{sitatan}$ is a type having three argument roles: the argument $[\xi]$ that can be filled up by an animate individual; the argument $[\dot{s}']$ can be filled up only by a situation where the individual filling up the role $[\xi]$ is animate; and $[\dot{l}']$ can be filled up only by the corresponding location in that situation. E.g., as in the proposition (48b) that is constituent of the type $T_{sitotutu}$ in (48a)–(48c).

The type $T_{sitotutu}$ in (48a)–(48c) is a three-argument type, with argument roles for an individual, $\text{IND} : [x]$, a situation, $\text{SIT} : [\dot{s}]$, and a location, $\text{LOC} : [\dot{l}]$, and is a type of odd-toed ungulate individuals x in a situation \dot{s} and a location \dot{l} :

$$T_{sitotutu} = [\text{IND} : [x], \text{SIT} : [\dot{s}], \text{LOC} : [\dot{l}]] \mid \quad (48a)$$

$$(T_{sitatan}, \text{IND} : [\xi] : x, \text{SIT} : [\dot{s}'] : \dot{s}, \text{LOC} : [\dot{l}'] : \dot{l}) \wedge \quad (48b)$$

$$(\dot{s} \models \ll \text{odd-toed}, \text{IND} : \text{arg} : x, \text{LOC} : \dot{l}; 1 \gg \wedge \quad (48c)$$

$$\ll \text{ungulate}, T_{sitatan} : \text{arg} : x, \text{LOC} : \dot{l}; 1 \gg)]$$

The constituent proposition $(T_{sitatan}, \text{IND} : [\xi] : x, \text{SIT} : [\dot{s}'] : \dot{s}, \text{LOC} : [\dot{l}'] : \dot{l})$ in (48b) states that the objects x , \dot{s} , and \dot{l}

(filling correspondingly the argument-roles $[\xi]$, $[\dot{s}']$, and $[\dot{l}']$ of the type $T_{sitatan}$) have to be of type $T_{sitatan}$.

B. Interrelated Generalizations

Now, we demonstrate the generalization of the type $T_{nocturnal}$ in (37a) to a situated type that has three arguments:

$$T_{sitnoct} = [\text{IND} : [\zeta], \text{SIT} : [\dot{s}''], \text{LOC} : [\dot{l}'']] \mid \quad (49a)$$

$$(\dot{s}'' \models \ll \text{nocturnal}, \text{IND} : \zeta, \text{LOC} : \dot{l}''; 1 \gg)] \quad (49b)$$

The type $T_{sitnoctotutu}$ in (50a)–(50e) is a three-argument type, with argument roles for an individual, $\text{IND} : [x]$, a situation, $\text{SIT} : [\dot{s}]$, and a location, $\text{LOC} : [\dot{l}]$, and is a type of odd-toed ungulate individuals x , that are nocturnal, in a situation \dot{s} and a location \dot{l} .

As in (48a)–(48c), the constituent proposition $(T_{sitatan}, \text{IND} : [\xi] : x, \text{SIT} : [\dot{s}'] : \dot{s}, \text{LOC} : [\dot{l}'] : \dot{l})$ in (50b) states that the object parameters x , \dot{s} , and \dot{l} (filling correspondingly the argument-roles $[\xi]$, $[\dot{s}']$, and $[\dot{l}']$ of the type $T_{sitnoct}$) have to be of type $T_{sitnoct}$. Any objects respectively instantiating these parameters have to be of these types too. The constituent $(T_{sitnoct}, \text{IND} : [\zeta] : x, \text{SIT} : [\dot{s}''] : \dot{s}, \text{LOC} : [\dot{l}''] : \dot{l})$ in (50c) is a proposition, which stating that the same object parameters x , \dot{s} , and \dot{l} (filling correspondingly the argument-roles $[\zeta]$, $[\dot{s}'']$, and $[\dot{l}'']$) have to be of type $T_{sitnoct}$, also.

$$T_{sitnoctotutu} = [\text{IND} : [x], \text{SIT} : [\dot{s}], \text{LOC} : [\dot{l}]] \mid \quad (50a)$$

$$(T_{sitatan}, \text{IND} : [\xi] : x, \text{SIT} : [\dot{s}'] : \dot{s}, \text{LOC} : [\dot{l}'] : \dot{l}) \wedge \quad (50b)$$

$$(T_{sitnoct}, \text{IND} : [\zeta] : x, \text{SIT} : [\dot{s}''] : \dot{s}, \quad (50c)$$

$$\text{LOC} : [\dot{l}''] : \dot{l}) \wedge$$

$$(\dot{s} \models \ll \text{odd-toed}, \text{IND} : \text{arg} : x, \text{LOC} : \dot{l}; 1 \gg \wedge \quad (50d)$$

$$\ll \text{ungulate}, T_{sitatan} : \text{arg} : x, \text{LOC} : \dot{l}; 1 \gg)] \quad (50e)$$

Therefore, the truth conditions for the proposition (51a) are equivalent to those for the conjunction (51b)–(51d), where a , s , and l , are either specific objects or parameters, respectively for an individual, situation, and space-time location. I.e., the proposition (51a) stating that a , s , l are of the type $T_{sitnoctotutu}$ is true iff the conjunctive proposition (51b)–(51d) is true by instantiating the parameters x , \dot{s} , \dot{l} with a , s , l , correspondingly:

$$(T_{sitnoctotutu}, \text{IND} : [x] : a, \text{SIT} : [\dot{s}] : s, \text{LOC} : [\dot{l}] : l) \quad (51a)$$

\iff

$$(T_{sitatan}, \text{IND} : [\xi] : x, \text{SIT} : [\dot{s}'] : \dot{s}, \text{LOC} : [\dot{l}'] : \dot{l}) \wedge \quad (51b)$$

$$(T_{sit-nocturnal}, \text{IND} : [\zeta] : x, \text{SIT} : [\dot{s}''] : \dot{s}, \quad (51c)$$

$$\text{LOC} : [\dot{l}''] : \dot{l}) \wedge$$

$$(\dot{s} \models \ll \text{odd-toed}, \text{IND} : \text{arg} : x, \text{LOC} : \dot{l}; 1 \gg \wedge \quad (51d)$$

$$\ll \text{ungulate}, T_{sitatan} : \text{arg} : x, \text{LOC} : \dot{l}; 1 \gg)] \quad (51e)$$

$$\text{is true for } c(x) = a, c(\dot{s}) = s, c(\dot{l}) = l, \quad (51e)$$

where c is a function for parameter assignment, i.e., instantiating, the parameters x , \dot{s} , \dot{l} with a , s , l , correspondingly. In what follows, we shall use simply the equality sign for direct parameter assignment, without using the assignment function

c. The above information can be expressed all together in the following way:

$$(T_{sitnoctotu}, \text{IND} : [x] : a, \text{SIT} : [\dot{s}] : s, \text{LOC} : [\dot{l}] : l) \quad (52a)$$

$$\iff (p_1 \wedge p_2 \wedge p_3) \text{ is true} \quad (52b)$$

where

$$p_1 = ([x], [\dot{s}], [\dot{l}] \mid (T_{sitatan}, \text{IND} : [\xi] : x, \text{SIT} : [\dot{s}'] : \dot{s}, \text{LOC} : [\dot{l}'] : \dot{l})), \quad (53a)$$

$$\text{IND} : [x] : a, \text{SIT} : [\dot{s}] : s, \text{LOC} : [\dot{l}] : l), \quad (53b)$$

$$p_2 = ([x], [\dot{s}], [\dot{l}] \mid (T_{sitnoct}, \text{IND} : [\zeta] : x, \text{SIT} : [\dot{s}'] : \dot{s}, \text{LOC} : [\dot{l}'] : \dot{l})), \quad (53c)$$

$$\text{IND} : [x] : a, \text{SIT} : [\dot{s}] : s, \text{LOC} : [\dot{l}] : l), \quad (53d)$$

$$p_3 = ([x], [\dot{s}], [\dot{l}] \mid (\dot{s} \models \ll \text{odd-toed}, \text{IND} : \text{arg} : x, \text{LOC} : \dot{l}; 1 \gg \wedge \ll \text{ungulate}, T_{sitatan} : \text{arg} : x, \text{LOC} : \dot{l}; 1 \gg)), \quad (53e)$$

$$\text{IND} : [x] : a, \text{SIT} : [\dot{s}] : s, \text{LOC} : [\dot{l}] : l) \quad (53f)$$

Now, instead of repeating the informational patterns in (52b)–(53f), for various specific types and properties, we can use sub-typing, by type and relation parameters, which can be instantiated as necessary.

$$(T_{sitnoctotu}, \text{IND} : [x] : a, \text{SIT} : [\dot{s}] : s, \text{LOC} : [\dot{l}] : l) \quad (54a)$$

$$\iff (p_1 \wedge p_2 \wedge p_3) \quad (54b)$$

where

$$p_1 = ([x], [\dot{s}], [\dot{l}] \mid (T_1, \text{IND} : [\xi] : x, \text{SIT} : [\dot{s}'] : \dot{s}, \text{LOC} : [\dot{l}'] : \dot{l})), \quad (55a)$$

$$\text{IND} : [x] : a, \text{SIT} : [\dot{s}] : s, \text{LOC} : [\dot{l}] : l),$$

$$p_2 = ([x], [\dot{s}], [\dot{l}] \mid (T_2, \text{IND} : [\zeta] : x, \text{SIT} : [\dot{s}'] : \dot{s}, \text{LOC} : [\dot{l}'] : \dot{l})), \quad (55b)$$

$$\text{IND} : [x] : a, \text{SIT} : [\dot{s}] : s, \text{LOC} : [\dot{l}] : l),$$

$$p_3 = ([x], [\dot{s}], [\dot{l}] \mid (\dot{s} \models \ll r, \text{IND} : [x] : x, \text{LOC} : [l] : l; 1 \gg)), \quad (55c)$$

$$\text{IND} : [x] : a, \text{SIT} : [\dot{s}] : s, \text{LOC} : [\dot{l}] : l),$$

$$r = ([x], [\dot{l}] \mid \ll r_u, \text{IND} : [x] : x, \text{LOC} : [l] : l; 1 \gg \wedge \ll r_t, \text{IND} : [x] : x, \text{LOC} : [l] : l; 1 \gg]), \quad (55d)$$

$$r_u = [\text{IND} : [x], \text{LOC} : [\dot{l}] : \dot{l} \mid \ll u, T_1 : \text{arg} : x, \text{LOC} : \dot{l}; 1 \gg], \quad (55e)$$

$$r_t = [\text{IND} : [x], \text{LOC} : [\dot{l}] : \dot{l} \mid \ll t, \text{IND} : \text{arg} : x, \text{LOC} : \dot{l}; 1 \gg], \quad (55f)$$

$$T_1 = T_{sitatan}, T_2 = T_{sitnoct}, u = \text{ungulate}, \quad (55g)$$

$$t = \text{odd-toed}, \quad (55h)$$

$$a = a_0, s = s_0, l = l_0 \quad (55i)$$

Now, the parameter t in (54b)–(55i) can be left under-specified, by dropping out the instantiation $t = \text{odd-toed}$ in (55h). However, one may still need to restrict the possible instantiations of t . E.g., that can be done by replacing (55f) and (55h) with (56a)–(56b).

$$r_t = [\text{IND} : [x], \text{LOC} : [\dot{l}] : \dot{l} \mid \ll t_o, \text{IND} : \text{arg} : x, \text{LOC} : \dot{l}; 1 \gg \vee \ll t_e, \text{IND} : \text{arg} : x, \text{LOC} : \dot{l}; 1 \gg], \quad (56a)$$

$$t_o = \text{odd-toed}, t_e = \text{even-toed} \quad (56b)$$

Another possibility to introduce alternative instantiations, i.e., alternative parameter assignments, is by using sets and membership instantiations:

$$p \in M, \text{ for given } T : \text{TYPE}, p \in \text{PAR}, p : T, \text{ and a set } M \text{ of objects of type } T \quad (57)$$

I.e., in (54b)–(55i), we can replace the assignment $t = \text{odd-toed}$ in (55h) with (58).

$$t \in \{ \text{odd-toed}, t = \text{even-toed} \} \quad (58)$$

The truth values of the proposition in (51a), i.e., in (52a) and (54a), can be determined, i.e. calculated, from (51b)–(51e), alternatively from (52b)–(53f), or (54b)–(55h). The informational structures in (52b)–(53f) and in (54b) reveal detailed informational compounds and how they are “linked” in general informational patterns. In particular, the informational structure of (54b) reveals:

- 1) The informational structures of the propositions p_1 and p_2 have the same pattern given in (59):

$$(T, \text{IND} : [\xi] : x, \text{SIT} : [\dot{s}'] : \dot{s}, \text{LOC} : [\dot{l}'] : \dot{l}) \quad (59)$$

Note that the distinctions between \dot{s}' and \dot{s}'' , and respectively, between \dot{l}' and \dot{l}'' , are inessential. The propositions p_1 and p_2 differ in the specific instantiation of the type parameter T with $T_{sitatan}$ and $T_{sitnoct}$, respectively.

- 2) p_3 is the general information pattern for a situated, propositional content, where r is a relation parameter, which can be instantiated, i.e., anchored to by a parameter assignment, or equality like (55d), to any unary relation (without counting the location argument, which is specially designated). In this case it is instantiated by a complex conjunction infon. The parameter r can be considered as a relation parameter r with any number of arguments, $n \geq 0$:

$$(\dot{s} \models \ll r, T_1 : \text{arg}_1 : x_1, \dots, T_1 : \text{arg}_n : x_n, \text{LOC} : [l] : l; 1 \gg) \quad (60)$$

VIII. GENERAL ARGUMENT STRUCTURE IN COMPLEX RELATIONS

The relation parameters r_t and r_u are instantiated to complex relations by (55f) and (55e) respectively, both of which have the same informational structure as r_{2,R_2} in (61):

$$r_{2,R_2} = [\text{IND} : [x], \text{LOC} : [\dot{l}] : \dot{l} \mid \ll R_2, \text{IND} : \text{arg} : x, \text{LOC} : \dot{l}; 1 \gg] \quad (61)$$

The relation r_{m,R_n} , in (62), is the generalization of the two-argument, parametric relation r_{2,R_2} in (61) to a parametric relation r_{m,R_n} of several complex argument roles $[x_{j_1}], \dots, [x_{j_m}]$, $[\dot{l}]$, by abstraction over the location parameter \dot{l} , and a set of parameters, $T_{j_1} : x_{j_1}, \dots, T_{j_m} : x_{j_m}$, filling some of the argument roles of an n -argument relation parameter R_n :

$$r_{m,R_n} = [T_{j_1} : [x_{j_1}], \dots, T_{j_m} : [x_{j_m}], \text{LOC} : [\dot{l}] : \dot{l} \mid \ll R_n, T_1 : \text{arg}_1 : x_1, \dots, T_n : \text{arg}_n : x_n, \text{LOC} : \dot{l}; 1 \gg] \quad (62)$$

IX. CONCLUSIONS AND FUTURE WORK

A. Conclusions

The paper covers the topic of linking parametric information in taxonomic concepts that are constructed by using relations and types having complex argument roles. In general, the argument roles of complex relations and types are constrained by situational types. The argument roles can only be saturated by appropriate situational objects of the corresponding types. The objects that fill-up the complex argument roles in parametric objects can be parametric too. Constituents of information can be linked via recursively constructed, parametric components and instantiations.

We have shown how informational patterns can be productively re-used by different instantiations, with updated situational constituents, which can be specific or again parametric. The constraints over abstract argument roles are expressed recursively with complex types, via abstraction operator.

The presented situation-theoretic approach is primarily intended for modeling information that is typically partial, parametric, and depends on context and situations, as in concepts. In addition, in the contemporary explosion of technologies for large databases of uncertain information, we find that Situation Theory can be used for efficient representation of large collections of data and information streams. Information can be hierarchically organized according to informational types in classes and subclasses. Instead of repeating larger or smaller amounts of details, concepts represent parametric generalizations, which can be instantiated depending on needs and specific situations. Typically, parametric instantiations are dependent on situations. Situation-theoretic types, which are used in concepts with components that are parameters for situations, carry informational content that is “placed” in abstract, “unknown” situations.

B. Future Developments

Future work, which is related to the topic of this paper, is linking parametric objects via complex restricted parameters, for representing underspecified cognitive concepts. Situation Theory with similar parametric objects has been used for semantics of attitude expressions and quantifier ambiguities (e.g., see Loukanova [11], [12], [22]).

A related topic, on which we work concurrently, is development of formal languages for Situation Theory, see Loukanova [10], [21]. Forthcoming work is on association of such formal languages with denotational and algorithmic semantics. The denotational structures of such formal languages are situation-theoretic domains, as in Loukanova [7]–[9], and in this paper. The varieties of formal languages depend on the areas of application, coverage of semantic concepts, and the situation-theoretic domains of objects. A formal language of Situation Theory can be useful for expressing general semantic concepts such as the ones introduced in this paper.

As we explained in Section I, Aczel non-well-founded set theory provides a theoretic limitation of the situation-theoretic objects, including situations, to objects that conform to Aczel Anti-Foundation Axiom (AFA). These objects can be large as non-well-founded sets, and can be circular, but they have finite representations, e.g., by finite graphs, which, in case of circular information, are circular graphs. Specific applications can have additional restrictions on the objects in their domain and still allow circular information, if needed. In typical applications, circularity is undesirable by leading to circular algorithms, which may not end. Excluding circular situation-theoretic objects from the domains is complicated at model-theoretic level. Using a formal language of Situation Theory, as in Loukanova [10], provides a useful tool by formulation of acyclicity constraints over formal terms denoting situation-theoretic objects.

A primary area of applications of Situation Theory is to computational semantics of languages. Typical approaches to computational semantics encounter problems due to progressive expansion of ambiguities, and multiple, unknown, or undetermined interpretations, which actually depend on partial, and underspecified semantic information. Situation-theoretic objects, as introduced in this paper, model informational content of concepts by using semantic parameters, which can be instantiated depending on context, situations, events, and other resources.

Among the target areas of applications of the presented approach are neuroscience of language and cognitive science, where cognitive concepts are essential. Of particular interests are situation-theoretic models of forming, developing, productivity, and efficiency of language and cognitive concepts and universals.

An open area of work is on formal languages of Situation Theory, e.g., as in Loukanova [10], [21], expanding them with syntax-semantics inferences, checks for consistency, and in addition, implementing them as computerized systems. Such practical implementations require work on developments

of specialized algorithms, their classification with respect to complexity, developments of databases, and techniques for evaluations.

C. Comparative Studies

Detailed comparative study of the technique introduced in this paper with other techniques for representation of concepts is an unexplored area. The technique introduced here shares ideas with work from fuzzy networks, see Yager [23]. Another direction of comparative study, by considering prospects for new developments, is with the approach of Rough Neural Computing (RNC), see Pal et al. [24]. Comparison with RNC would be of especial interest because RNC is a development of modeling neural networks for computations by using lexemes from human language. Similarly, situation-theoretic representations of concepts, as introduced here, have essential components consisting of semantic representations of lexical items and phrases from human language. Comparative studies may open possibilities for enhanced integration of approaches. We have a preliminary view that approaches that use fuzzy logic and fuzzy sets can benefit by further developments integrating Situation Theory and formal languages for it. This will enhance representation of fuzzy informational concepts and other informational units, by enriching them with content that is structured information, even if partly known and otherwise underspecified. On the other hand, Situation Theory and formal languages for it, may benefit by incorporation of fuzzy set theory and weighted parametric information, for practical applications.

A promising direction of developments is for relational databases with Situation Theory and semantic representations, by using formal languages of it. This is essential for domains such as health and medical sciences, where semantic information is typically partial, underspecified, and dependent on situations. E.g., situation-theoretic approach introduced here can contribute to enhanced databases systems introduced in Ślęzak et al. [25], and Sosnowski and Ślęzak [26].

REFERENCES

- [1] J. Barwise, "Scenes and other situations," *Journal of Philosophy*, vol. 78, pp. 369–397, 1981.
- [2] J. Barwise and J. Perry, *Situations and Attitudes*. Cambridge, MA: MIT press, 1983, republished as [27].
- [3] P. Aczel, *Non-well-founded Sets*, ser. CSLI Lecture Notes. Stanford, California: CSLI Publications, 1988, vol. 14.
- [4] M. Rathjen, "Predicativity, circularity, and anti-foundation," in *One hundred years of Russell's paradox (De Gruyter Series in Logic and Its Applications)*, G. Link, Ed. Walter de Gruyter, Berlin, New York, 2004, vol. 6, pp. 191–219.
- [5] R. H. Thomason, Ed., *Formal Philosophy: Selected Papers of Richard Montague*. New Haven, Connecticut: Yale University Press, 1974.
- [6] K. Devlin, "Situation theory and situation semantics," in *Handbook of the History of Logic*, D. Gabbay and J. Woods, Eds. Elsevier, 2008, vol. 7, pp. 601–664.
- [7] R. Loukanova, "Situating Propositions with Constraints and Restricted Parameters," in *Proceedings of the 6th International Workshop on Constraints and Language Processing*, ser. Computer Science Research Reports, P. Blache, H. Christiansen, V. Dahl, and J. Villadsen, Eds., no. 134, October 2011, pp. 44–55.
- [8] —, "Situating Agents in Linguistic Contexts," in *Proceedings of the 5th International Conference on Agents and Artificial Intelligence*, J. Filipe and A. Fred, Eds., vol. 1. Barcelona, Spain: SciTePress — Science and Technology Publications, 2013, pp. 494–503.
- [9] —, "Situation Theory, Situated Information, and Situated Agents," in *Transactions on Computational Collective Intelligence XVII*, ser. Lecture Notes in Computer Science, N. T. Nguyen, R. Kowalczyk, A. Fred, and F. Joaquim, Eds. Springer Berlin Heidelberg, 2014, vol. 8790, pp. 145–170.
- [10] —, "Underspecified Relations with a Formal Language of Situation theory," in *Proceedings of the 7th International Conference on Agents and Artificial Intelligence*, S. Loiseau, J. Filipe, B. Duval, and J. van den Herik, Eds., vol. 1. SCITEPRESS — Science and Technology Publications, Lda., 2015, pp. 298–309.
- [11] —, "Quantification and Intensionality in Situation Semantics," in *Computational Linguistics and Intelligent Text Processing*, ser. Lecture Notes in Computer Science, A. Gelbukh, Ed. Springer Berlin / Heidelberg, 2002, vol. 2276, pp. 32–45.
- [12] —, "Generalized Quantification in Situation Semantics," in *Computational Linguistics and Intelligent Text Processing*, ser. Lecture Notes in Computer Science, A. Gelbukh, Ed. Springer Berlin / Heidelberg, 2002, vol. 2276, pp. 46–57.
- [13] C. Pollard and I. A. Sag, *Information-Based Syntax and Semantics, Part I*, ser. CSLI Lecture Notes. CSLI Publications, 1987, no. 13.
- [14] —, *Head-driven Phrase Structure Grammar*. Chicago, IL: University of Chicago Press, 1994.
- [15] I. A. Sag, T. Wasow, and E. M. Bender, *Syntactic Theory: A Formal Introduction*. Stanford, California: CSLI Publications, 2003.
- [16] A. Copestake, D. Flickinger, C. Pollard, and I. Sag, "Minimal recursion semantics: an introduction," *Research on Language and Computation*, vol. 3, pp. 281–332, 2005.
- [17] R. Loukanova, "From Montague's Rules of Quantification to Minimal Recursion Semantics and the Language of Acyclic Recursion," in *Biology, Computation and Linguistics — New Interdisciplinary Paradigms*, ser. Frontiers in Artificial Intelligence and Applications, G. Bel-Enguix, V. Dahl, and M. D. Jiménez-López, Eds. Amsterdam; Berlin; Tokyo; Washington, DC: IOS Press, 2011, vol. 228, pp. 200–214.
- [18] Y. N. Moschovakis, "A logical calculus of meaning and synonymy," *Linguistics and Philosophy*, vol. 29, pp. 27–89, 2006.
- [19] J. Ginzburg and I. A. Sag, *Interrogative Investigations: The Form, Meaning, and Use of English Interrogatives*. Stanford, California: CSLI Publications, 2000.
- [20] M. Van Lambalgen and F. Hamm, *The Proper Treatment Of Events*, ser. Explorations in Semantics. Oxford: Wiley-Blackwell, 2004.
- [21] R. Loukanova, "A Formalization of Generalized Parameters in Situated Information," (to appear).
- [22] —, "Russellian and Strawsonian Definite Descriptions in Situation Semantics," in *Computational Linguistics and Intelligent Text Processing*, ser. Lecture Notes in Computer Science, A. Gelbukh, Ed. Springer Berlin / Heidelberg, 2001, vol. 2004, pp. 69–79.
- [23] R. Yager, "Concept Representation and Database Structures in Fuzzy Social Relational Networks," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 40, no. 2, pp. 413–419, March 2010.
- [24] S. K. Pal, L. Polkowski, and A. Skowron, *Rough Neural Computing: Techniques for Computing with Words*. Springer, 2004.
- [25] D. Ślęzak, A. Janusz, W. Świeboda, H. S. Nguyen, J. G. Bazan, and A. Skowron, "Semantic analytics of PubMed content," in *Information Quality in e-Health*. Springer, 2011, pp. 63–74.
- [26] Ł. Sosnowski and D. Ślęzak, "How to Design a Network of Comparators," in *Brain and Health Informatics*, ser. Lecture Notes in Computer Science, K. Imamura, S. Usui, T. Shirao, T. Kasamatsu, L. Schwabe, and N. Zhong, Eds. Springer International Publishing, 2013, vol. 8211, pp. 389–398.
- [27] J. Barwise and J. Perry, *Situations and Attitudes*, ser. The Hume Series. Stanford, California: CSLI Publications, 1999.