Operationalization of the Blending and the Levels of Abstraction Theories with the Timed Observations Theory

Marc Le Goc[1] and Fabien Vilar[1,2]

[1] *Aix-Marseille University, Laboratory for Sciences of Information and Systems (LSIS), UMR CNRS 7296, Marseille, France*

[2] *TOM4, Salon de Provence, France (www.tom4web.com)*

Keywords: Conceptual Integration Networks, Conceptual Blending, Abstraction, Level of Abstraction, Gradient of Abstraction, Knowledge Engineering.

Abstract: Providing a meaning to observations coming from humans (interviews) or machines (data sets) is a necessity to build adequate analysis and efficient models that can be used to take a decision in a given domain. Fauconnier and Turner demonstrates in 1998 the cognitive power of their Blending Theory where the blending of multiple conceptual networks is presented as a general-purpose, fundamental, indispensable cognitive operation to this aim. On the other hand, Floridi proposed in 2008 a theory of levels of abstraction as a fundamental epistemological method of conceptual analysis that can also be used to this aim. Both theories complete together but both lack of mathematical foundations to build an operational data and knowledge modeling method that helps and guides the Analysts and the Modeling Engineers. In this theoretical paper, we introduce the mathematical framework, based on the Timed Observations Theory, designed to build a method of abstraction merging together the Blending Theory and the Levels of Abstraction Theory. Up to our knowledge, this is the first mathematical theory allowing the operationalization of the Blending Theory and the Levels of Abstraction Theory. All over the paper, the mathematical framework is illustrated on an oral exchange between three persons observing a vehicle. We show that this framework allows to build a rational meaning of this exchange under the form of a superposition of three abstraction levels.

1 INTRODUCTION

With the always increasing amount of data collected over things connected on information networks, the need for data and knowledge analysis became a crucial stake for most of the industrial and service activities, including the research activity itself. The main difficulty with data and knowledge analysis resides in the introduction, in a controlled way, of semantics in the syntactic patterns provided by human analysts with the eventual help of Statistic Learning or Data Mining algorithms. There is then a crucial need for models able to guide a rationale interpretation of data providing from humans or machines.

(Fauconnier and Turner, 1998) proposed the theory of *Conceptual Integration Networks*, also called the *Blending Theory*, that defines a common conceptual operation, the *blending of conceptual spaces*, to provide a meaning and a way to compress the representations that are useful for knowledge memorization and manipulation. Blending of different conceptual spaces *plays a fundamental role in the construction of meaning in everyday life, in the arts and sciences, and especially in the social and behavioral sciences* (Fauconnier and Turner, 2003). The essence of the conceptual blending operation is to establish a new conceptual space through the matching between the contain of different conceptual spaces. Fauconnier and Turner suggest that *the capacity for complex conceptual blending is the crucial capacity needed for thought and language* (Fauconnier and Turner, 2003).

Another but complementary point of view is proposed in (Floridi, 2008; **?**) to address the problem of defining the nature of natural, human or artificial agents with the notion of *Level of Abstraction*. The *Levels of Abstraction Theory* aims to clarify implicit assumptions and to allow the resolution of possible conceptual confusions with the comparison between different point of view about the same phenomenon (concrete or abstract). Similarly to the *Blending Theory*, it *provides a detailed and controlled way of comparing analyses and models* (Floridi, 2008) with the introduction of multiple levels of abstraction in conceptual analysis. It constitutes then a crucial and pow-

Le Goc M. and Vilar F.
Operationalization of the Blending and the Levels of Abstraction Theories with the Timed Observations Theory.
DOI: 10.5220/0006111103640373
In *Proceedings of the 9th International Conference on Agents and Artificial Intelligence (ICAART 2017)*, pages 364-373
ISBN: 978-989-758-220-2

erful tool to address the analysis and the modeling of the phenomenon under consideration. Floridi argued that *for discrete systems, whose observables take on only finitely-many values, the method is indispensable* (Floridi, 2008).

These two theories share common goals but develop different ways to achieve them, and both lack of mathematical foundations. The aim of this paper is to propose an adequate mathematical framework that provides for the first time, up to our knowledge, a strong formal foundation to these theories. The proposed mathematical framework, called Tom4A *Timed Observation Method for Abstraction*, is build on the *Timed Observations Theory* (TOT, (Le Goc, 2006)) and constitutes the basis of a new abstraction approach . Clearly, our long term goal is to develop software tools able to discover and to model knowledge representations from sets of timed data so that the human interpretation is intuitive, immediate and independent of the learning and the modeling tools.

To make the mathematical framework as simple and intuitive as possible, the main concepts are illustrated with a running example of three speakers discussing about a vehicle (cf. Section 2), the original text coming from the web site of the *Society for the Philosophy of Information* (http://www.socphilinfo.org/node/150). Section 3 provides the principles and the formal modeling tools of the TOT that will be used all along this paper. Section 4 describes the building of the conceptual spaces of the three speakers. Section 5 introduces the blending process to build the common model used by the speakers to understand together. Section 6 define the notion of generic conceptual space. This section ends the introduction of the basic modeling elements of the *Blenbing Theory*. Section 7 introduces the basis of the *Levels of Abstraction Theory* that will be used to add an inference structure to the blended and the generic conceptual spaces and to organize them with both a *disjoint* and a *nested* gradients of abstraction. This section shows also that the formalized notion of *gradient of abstraction* constitute a powerful tool to capture and to represent the meaning in a coherent and formal way. Finally, section 8 proposes a short synthesis of Tom4A, and provide some insights about our future works.

2 RUNNING EXAMPLE

In this section, only the factual elements of the running example are given in *verbatim*, its analysis according to the *Levels of Abstraction Theory* being available in http://www.socphilinfo.org/node/150:

Suppose we join Alice, Bob, and Carol earlier on at the party. They are in the middle of a conversation. We do not know the subject of their conversation, but we are able to hear this much:

- *Alice observes that its (whatever it is) old engine consumed too much, that it has a stable market value but that its spare parts are expensive;*
- *Bob observes that its engine is not the original one, that its body has been recently re-painted but that all leather parts are very worn;*
- *Carol observes that it has an anti-theft device installed, is kept garaged when not in use, and has had only a single owner.*

The point to notice in these elements is the fact the three speakers *observe* properties about an unknown, for us, *system*. The three speakers exchange then *observations* about the system. Since Aristotle, we know that this type of discourse can be resumed with a set of apophantic formulas of the form *Subject1-Copula-Subject2*. Each observation is then a *proposition*, that is to says a *relation* between two subjects, the nature of the relation being defined with a verb (the copula) linking the two subjects. For example, Alice's observation *it has a stable market value* is a proposition that can be represented with the binary predicate *is* linking the subject *MarketValue* and the other subject *stable*: $is(MarketValue, stable)$. So, the exchange can be re-written to make clear the different observations of each speakers:

1. Alice: its engine is old.
2. Alice: its engine consumed too much.
3. Alice: it has a stable market value.
4. Alice: its spare parts are expensive.
5. Bob: its engine is not the original one.
6. Bob: its body has been recently re-painted.
7. Bob: all leather parts are very worn.
8. Carol: its anti-theft device is installed.
9. Carol: it is kept garaged when not in use.
10. Carol: it has had only a single owner.

Each but the 9^{th} proposition can be easily formalized with a binary predicate. Because of the use of the *when* connector, the observation 9 links two propositions: $is(it, not\_in\_use)$ and $is\_kept(it, garaged)$. Yet, numerous of these observations use verbs conjugated to the past, meaning that the observations have a time reference. Temporal versions of the first order Predicate Logic would then be used to formalize such observations but the interpretation of the resulting formulas is accessible for only specialists of these logics.

To keep an intuitive interpretation of the formulas, Tom4A uses the modeling principles of the Timed Observations Theory that are introduced in the next section. These principles will be applied (i) to build the conceptual space of the three speakers, (ii) to define the corresponding blended and generic spaces and (iii) to model the blended space with a Gradient of Abstraction (GoA) according to (Floridi, 2008).

3 MODELING WITH THE TOT

The Timed Observations Theory (TOT) provides a mathematical framework to model dynamic processes from timed data. The TOT is currently the mathematical basis of the TOM4L (Timed Observation Mining for Learning) Knowledge Discovering from Databases process (Le Goc et al., 2015; **?**), and the TOM4D (Timed Observation Modeling for Diagnosis) Knowledge Engineering methodology (Pomponio and Le Goc, 2014). Tom4A aims at introducing levels of abstraction and generic spaces (Le Goc and Gaeta, 2004) in TOM4L and Tom4D according to the notion of *conceptual equivalence* (Zanni et al., 2006).

The aim of the TOT is to model an *observed process* defined as a couple $(X(t), \Theta(X,\Delta))$ where $X(t)$ is an arbitrarily constituted set $X(t) = \{x_1(t), ..., x_{n_X}(t)\}$ of n_X timed functions $x_i(t)$ of continuous time t (the dynamic process), $X = \{x_1, x_2, ..., x_{n_X}\}$ is the set of the n_X variable names x_i corresponding to each time functions $x_i(t)$ and $\Theta(X,\Delta)$ is an *observation program* implemented in a human or a computer, the set $\Delta = \{\delta_j\}$ being a set of *constant* values. A dynamic process $X(t)$ is said to be *observed* by a program $\Theta(X,\Delta)$ when this latter aims at writing *timed observations* describing the *modifications* over time of the functions $x_i(t)$ of $X(t)$:

Definition 1. *Timed Observation*
Let $\Gamma = \{t_k\}_{t_k \in \Re}$ *be a set of arbitrary timestamps* t_k *at which* $\Theta(X,\Delta)$ *observes a time function* $x_i(t) \in X(t)$ *and* $\theta(x_\theta, \delta_\theta, t_\theta)$ *be a predicate implicitly implemented in* $\Theta(X,\Delta)$*;*

A timed observation $(\delta_j, t_k) \in \Delta \times \Gamma$ *made on* $x_i(t)$ *is the assignation of the values* x_i*,* δ_j *and* t_k *to the predicate* $\theta(x_\theta, \delta_\theta, t_\theta)$ *such that* $\theta(x_i, \delta_j, t_k)$*.*

For example, Alice's observation *it has a stable market value* is represented with the timed observation $(stable, t_k)$, t_k being the (unknown) timestamps of the instant where Alice pronounces this sentence during the conversation. So, Alice play the role of the observation program $\Theta(X,\Delta)$ and the assigned ternary predicate $is(MarketValue, Stable, t_k)$, corresponding to $\theta(x_i, \delta_j, t_k)$, provides a *meaning* to the timed observation $(stable, t_k)$. The explicit link between a variable x_i (*MarketValue*) and a constant δ_j (*stable*) is made with the notion of *observation class*:

Definition 2. *Observation Class*
Let $X = \{x_i\}_{i=1...n_{n_X}}$ *be the set of variable names corresponding to* $X(t)$ *and* $\Delta = \{\delta_j\}$ *a set of constant values an observation program* $\Theta(X,\Delta)$ *can use.*

An observation class $O_k = \{..., (x_i, \delta_j), ...\}$ *for* $\Theta(X,\Delta)$ *is a subset of* $X \times \Delta$*.*

Any association establishing a mapping $\Delta \mapsto X$ for each δ_j of Δ can be made. The simplest way, and the most used, to define observation classes is the use of singletons $O_j = \{(x_i, \delta_j)\}$ where the pair (x_i, δ_j) is the unique element the set O_j. For example, the observation class $O_s^A = \{(MarketValue, stable)\}$ has been implicitly used by Alice to reason about the system (i.e. *it*). It is then obvious that doing so, all but the observation 9 (*it is kept garaged when not in use*) are occurrences of a particular observation class, the observations 9 linking together two occurrences of two different observation classes:

1. Alice, *its engine is old*:
   $O_4^A(t_1) \equiv (old, t_1)$, $O_4^A = \{x_4^A, old\}$.
2. Alice, *its engine consumed too much*:
   $O_6^A(t_2) \equiv (too\_much, t_2)$, $O_6^A = \{x_6^A, too\_much\}$.
3. Alice, *it has a stable market value*:
   $O_2^A(t_3) \equiv (stable, t_3)$, $O_2^A = \{x_2^A, stable\}$.
4. Alice, *its spare parts are expensive*:
   $O_8^A(t_4) \equiv (expensive, t_4)$, $O_8^A = \{x_8^A, expensive\}$.
5. Bob, *its engine is not the original one*:
   $O_1^B(t_5) \equiv (original, t_5)$, $O_1^B = \{x_1^B, original\}$.
6. Bob, *its body has been recently re-painted*:
   $O_3^B(t_6) \equiv (recently, t_6)$, $O_3^B = \{x_3^B, recently\}$.
7. Bob, *all leather parts are very worn*:
   $O_5^B(t_7) \equiv (very\_worn, t_7)$, $O_5^B = \{x_5^B, very\_worn\}$.
8. Carol, *its anti-theft device is installed*:
   $O_1^C(t_8) \equiv (installed, t_8)$, $O_1^C = \{x_1^C, installed\}$.
9. Carol, *it is kept garaged when not in use*:
   $O_3^C(t_{10}) \equiv (garaged, t_{10})$, $O_3^C = \{x_3^C, garaged\}$, $O_2^C(t_{11}) \equiv (not\_in\_use, t_{11})$, $O_2^C = \{x_2^C, not\_in\_use\}$.
10. Carol, *it has had only a single owner*:
   $O_5^C(t_{12}) \equiv (single, t_{12})$, $O_5^C = \{x_5^C, single\}$.

Carol's observation number 9 defines two timed observations, $O_2^C(t_{11}) \equiv (not\_in\_use, t_{11})$ and $O_3^C(t_{10}) \equiv (garaged, t_{12})$, corresponding to two observation classes $O_2^C = \{x_2^C, not\_in\_use\}$ and $O_3^C = \{x_3^C, garaged\}$. The role of the timestamps t_k of a timed observation is to provide a temporal reference in a flow of observations. Carol's meaning of the term *when* being unclear, the timestamps allows to provide a meaning to it: the *when* can be interpreted as a reference to the past. In other words, the observation 9 can be interpreted as: *when it is not in use, it is kept garaged*. Such an interpretation entails that the status of the usage of *it* must be defined *before* the assertion of the location. The TOT defines the notion of *timed binary relation* to represent a *sequential* relation between two observations classes O_i and O_j:

Definition 3. *Temporal Binary Relation*
A temporal binary relation $r_{ij}(O_i, O_j, [\tau_{ij}^-, \tau_{ij}^+])$, $\tau_{ij}^- \in \Re$, $\tau_{ij}^+ \in \Re$, is an oriented relation between two observation classes O_i and O_j that is timed constrained with the $[\tau_{ij}^-, \tau_{ij}^+]$ interval.

This definition leads to define Carol's observation number 9 with the timed binary relation $r_{23}^C(O_2^C, O_3^C, [0, \tau_{23}^+])$, $\tau_{23}^- = 0$ meaning that the end of the usage of *it* can coincide with the beginning of the put in the garage (i.e. may be $t_{10} = t_{11}$). This example suffices to provide an intuitive comprehension of Tom4D's operational definition of *knowledge* (cf. (Pomponio and Le Goc, 2014) for a justification of this definition):

Definition 4. *Any relation logically consistent with a binary temporal relation of the form $r_{ij}(C_i, C_j, [\tau_{ij}^-, \tau_{ij}^+])$ is a piece of knowledge.*

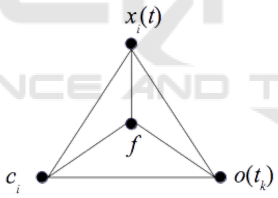

Figure 1: Basic Concepts of TOM4D models.

A model being an organized set of knowledge representations, the knowledge under consideration is a set of binary relations between time functions $x_i(t)$, constants δ_i and stochastic clocks Γ_i (cf. Figure 1).

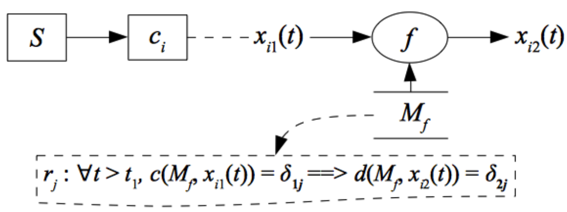

Figure 2: Tom4D's Representation of a Dynamic Function.

The notion of *dynamic function* plays a pivot role in the Tom4D modeling methodology. Figure 2 shows a graphical representation of the dynamic function $x_{i2}(t) = f(x_{i1}(t))$. The timed function $x_{i1}(t)$ is linked to a particular *component* c_i, itself being a part of the *container* of all the components, i.e. the *system* S. The dynamic function $x_{i2}(t) = f(x_{i1}(t))$ is defined over the Cartesian product $\Delta_{x_{i1}} \times \Delta_{x_{i2}}$ of the definition domain of the timed functions $x_{i1}(t)$ and $x_{i2}(t)$ respectively and implements a set of *decision rule* of the form:

$$\forall t \geq t_1, \forall \delta_{1j} \in \Delta_{x_{i1}}, \exists \delta_{2j} \in \Delta_{x_{i2}},$$
$$x_{i1}(t_1) = \delta_{1j} \xRightarrow{f} x_{i2}(t) = \delta_{2j}. \tag{1}$$

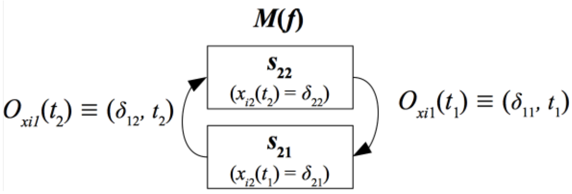

Figure 3: Finite State Machine Model of a Tom4D Function.

Such a set of decision rule specifies the *Finite State Machine* (FSM) of figure 3 constituting the *behavioral model* of the f dynamic function with $\Delta_{x_{i1}} = \{\delta_{11}, \delta_{12}\}$ and $\Delta_{x_{i2}} = \{\delta_{21}, \delta_{22}\}$. According to Tom4D, a rectangle represents a *discernible state* s_{ij} labeled with a proposition about the value of one or more functions at a particular timestamps, $x_{i2}(t_1) = \delta_{21}$ for s_{21} for example. An arrow represents a *transition* between two discernible states. Such a transition is conditioned with an occurrence of a particular observation class, $O_{x_{i1}}(t_1) \equiv (\delta_{11}, t_1)$ for example. This means that the dynamic function f implements the ternary predicate *equals* $= (x_i, \delta_j, t_k)$ of definition 1. In other words, the semantics of the ternary predicate $\theta(x_\theta, \delta_\theta, t_\theta)$ of definition 1 is given by the following two simple decision rules:

$$r_1 : \forall t \geq t_1, x_{i1}(t_1) = \delta_{11} \Longrightarrow x_{i2}(t) = \delta_{21}$$
$$r_2 : \forall t \geq t_2, x_{i1}(t_2) = \delta_{12} \Longrightarrow x_{i2}(t) = \delta_{22} \tag{2}$$

Clearly, with boolean sets, such a FSM is not necessary, these two basic decision rules are sufficient. But generally speaking, as figure 2 shows, a Tom4D dynamic function $x_2(t_k) = f(x_1(t_k))$ implements a *decision model* M_f specifying a set of decision rules linking the evaluation of a *criterion* $c(M_f, x_{i1}(t_1))$ about the value of a variable x_{i1} at time t_1 to a *decision* $d(M_f, x_{i2}(t))$ about the value of another variable x_{i2} at a posterior timestamps (or the same but not before): $r_j : \forall t \geq t_1, c(M_f, x_{i1}(t_1)) = \delta_{1j} \Longrightarrow d(M_f, x_{i2}(t)) = \delta_{2j}$. This formalism is necessary and sufficient to provide a formal meaning to the 11 observations of the speakers, and to build a semantic model of this exchange.

4 SPEAKERS' CONCEPTUAL SPACE

According to (Fauconnier and Turner, 1998), *mental spaces are small conceptual packets constructed as we think and talk, for purposes of local understanding and action.* They are *very partial assemblies containing elements, and structured by frames and cognitive models. They are interconnected, and can be modified as thought and discourse unfold.*

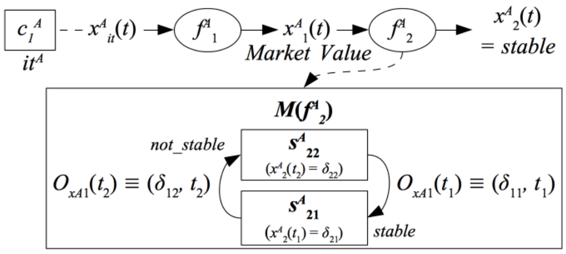

Figure 4: Observation Number 3 (Alice).

To apply this notion, let us consider again Alice's observation *it has a stable market value*. According to the Tom4D methodology, this observation can be formalized with a *dynamic function* $x_2^A(t) = f_2^A(x_1^A(t))$ (cf. figure 4) where $x_1^A(t)$ is the *time function* representing the *Market Value* evolution over time, and $x_2^A(t)$ is the *time function* representing Alice's assessments about the *Market Value*. The *variable* x_1^A denotes then Alice's *Market Value* concept. At the particular instant she is speaking, Alice's evaluation of the evolution of $x_1^A(t)$ is *stable*: she assigns then the value *stable* to the variable x_2^A. By construction, the definition domain of the variable x_2^A is then *at least* a boolean set $\Delta_{x_2^A} = \{$ *stable*, *not_stable* $\}$, the constant *not_stable* meaning anything but *stable*. This justifies the two states FSM implemented in the f_2^A assessment function.

The definition domain of the variable x_1^A is unknown. Nevertheless, if we interpret the concept of the *Market Value* with a usual dictionary, we can deduce that the dimension of x_1^A is an amount of money in a particular currency. This means that the definition domain of x_1^A is the set N of the natural numbers representing a number of cents in the implicit currency: $x_1^A \in N$. In other words, Alice's assessment function f_2^A is defined over the Cartesian product $N \times \Delta_{x_2^A}$. Now, clearly, the values of the variable x_1^A *must be* provided by a dynamic *measurement* function $x_1^A(t) = f_1^A(x_{it}^A(t))$. Such a function is either implemented in Alice's mind or, more surely, Alice make an implicit reference to an external function aiming at providing the *Market Value* of Alice's system it^A. This explains the relation, denoted with a dotted line, between the *component* labeled c_1^A and the variable x_{it}^A of figure 4. This component representing the term *it* in Alice's observation *it has a stable market value*, it formalizes Alice's notion of the system about which she talks. The component c_1^A is then the container of all the components of the system.

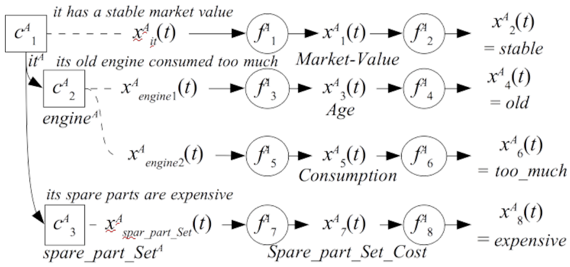

Figure 5: Alice's Conceptual Space.

Doing so for its four observations, it is simple to build a formal model of Alice's conceptual space as given in figure 5. This figure shows that Alice's observations concerned a system c_1^A made of two components c_2^A and c_3^A representing respectively Alice's notion of *engine* and *spare parts*. Two time functions are linked with the component c_2^A: $x_{engine1}^A(t)$ which is the input of the dynamic function f_3^A that *counts* the *age* of c_2^A, and $x_{engine2}^A(t)$, the input of f_5^A that *measure* the *consumption* of c_2^A. The time function $x_7^A(t) = f_7^A(x_{spare\_parts\_Set}^A(t))$ is a measurement function similar to f_1^A, the dynamic functions f_4^A, f_6^A and f_8^A being assessment functions similar to f_2^A. Obviously, these assessment functions use different decision models (the decision rules haven't been represented to simplify the figure 5).

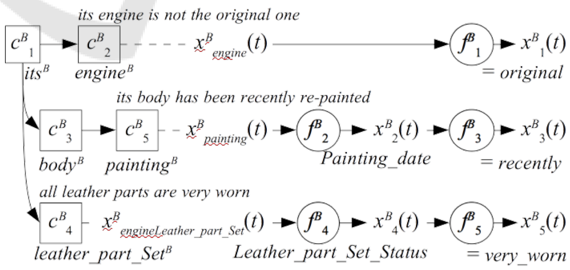

Figure 6: Bob's Conceptual Space.

Figure 6 shows Bob's conceptual space that has been made with the same method. To understand Bob's observations, a conceptual space made with two assessment functions, $x_3^B(t) = f_3^B(x_2^B(t))$ and $x_5^B(t) = f_5^B(x_4^B(t))$, must be built where $x_2^B(t)$ represents the current painting timestamps of the system's *body* and $x_4^B(t)$ the status of the *leather parts*. The function $x_1^B(t) = f_1^B(x_{engine}^B(t))$ is an *assertion* function allowing Bob to assert the *original* status of what

Bob names the *engine* c_2^B. An assertion function *can* directly provide a fact (or a property) from a time function, at the opposite of an assessment function which *must* operate on the values computed with a measurement function as f_4^B for f_5^B. The dynamic function $x_2^B(t) = f_2^B(x_{painting}^B(t))$ is a *dating* function that provides the timestamps of the most recent painting of the system body. Finally, Bob's notion of the system is the following set of components: $C^B = \{c_1^B, c_2^B, c_3^B, c_4^B, c_5^B\}$, c_5^B being linked with c_3^B.

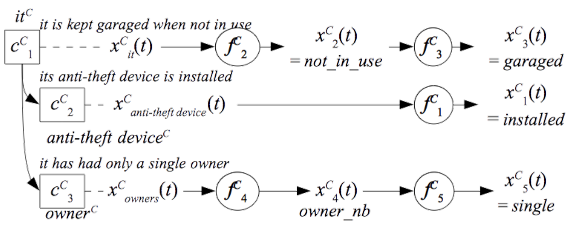

Figure 7: Carol's Conceptual Space.

Similarly, the same method leads to Carol's conceptual space of figure 7. The interpretation of Carol's observation number 9 (*when it is not in use, it is kept garaged*) leading to the timed binary relation $r_{23}^C(O_2^C, O_3^C, [0, \tau_{23}^+])$, it is represented with two successive assertion functions: the first, $x_2^C(t) = f_2^C(x_{it}^C(t))$ asserts the status of the *usage* of the system c_1^C, the second, $x_3^C(t) = f_3^C(x_2^C(t))$, asserts the *location* of c_1^C according to the values of $x_2^C(t)$. The two others observations 8 and 10 of Carol (*its anti-theft device is installed* and *it has had only a single owner* respectively) are modeled with the assertion functions $x_1^C(t) = f_1^C(x_{\text{anti-theft devise}}^C(t))$ and $x_5^C(t) = f_5^C(x_4^C(t))$. The particularity of the observation 10 is that to assert that c_1^C has had only one owner, the function $x_5^C(t) = f_5^C(x_4^C(t))$ needs the computing of the owner number. This is then the role of the *counting* function $x_4^C(t) = f_4^C(x_{\text{owners}}^C(t))$.

The conceptual space of Alice, Bob and Carol are those built by each of these speakers to produce their observations. The building of a blended space is now required to understand together the 10 observations.

5 BLENDED CONCEPTUAL SPACE

Blending is the usual name of the *conceptual integration* operation aiming to *project* at least two different conceptual spaces into a third one, the blended conceptual space: *conceptual integration-like framing or categorization-is a basic cognitive operation that operates uniformly at different levels of abstraction and under superficially divergent contextual circumstances* (Fauconnier and Turner, 1998). To cite again Fauconnier and Turner, *Projection is the backbone of analogy, categorization, and grammar* and they consider that it is *an established and fundamental finding of cognitive science that structure mapping and metaphorical projection play a central role in the construction of reasoning and meaning*.

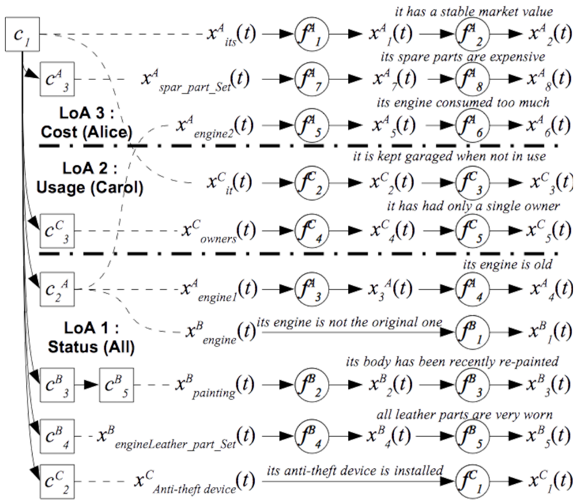

Figure 8: Blended Conceptual Space.

Since nothing is said in the example, we *must* make the following hypothesis to build a blended and a generic conceptual space: Alice, Bob and Carol speak about the same system. With this hypothesis, Tom4A's formalization principles make very simple the conceptual integration operation because the 10 observations are independent. Figure 8 shows the structure of the blended conceptual space. The exponents have been kept to clarify the links between the individual conceptual space and the resulting blended space after the projections of Alice's space firstly, next Bob's one and Carol's space lastly:

- The system is now represented with a unique component : $c_1 \equiv c_1^A \equiv c_1^B \equiv c_1^C$.
- The components of c_1 is the fusion of the Alice, Bob and Carol component sets: $C = \{c_1, c_2^A, c_2^C, c_3^A, c_3^B, c_3^C, c_4^B, c_5^B\}$.
- The time function's set is the fusion of the time function's sets: $X(t) = \{x_{its}^A(t), x_1^A(t), ..., x_1^C(t)\}$.
- The dynamic function's set is also the fusion of the corresponding sets: $F = \{f_1^A, ..., f_5^B, f_1^C\}$.

In figure 8, the 10 observations of the conversation have been organized in three abstraction levels. The lowest contains the more concrete observations of Alice, Bob and Carol: those concerning the *status* of the system c_1. The abstraction's level of the middle concerns its *usage* and contains Carol's observations number 9 and 10 only. The highest abstraction level concerns the *cost usage* of the system c_1. Only Alice made observations at this level of abstraction.

6 GENERIC CONCEPTUAL SPACE

One of the interesting features of Fauconnier and Turner's theory is the notion of *generic conceptual space*. This usual notion in the domain of Knowledge Engineering is of the main importance to model a knowledge corpus (cf. the CommonKads methodology (Schreiber et al., 2000) or (Pomponio and Le Goc, 2014) for a detailed illustration). Tom4D's Knowledge Engineering methodology allows to build generic conceptual spaces according to a notion of *conceptual equivalence* (Zanni et al., 2006) between different *knowledge roles*. *A knowledge role is an abstract label that indicates the role that the domain knowledge to which the label is attached plays in an inference process* (Bredeweg, 1994). So, the basic idea of the *conceptual equivalence* is that when two different concepts play the same role in a reasoning process, they can be considered as *conceptually equivalent*.

Let us consider together Alice's observation number 3 (*it has a stable market value*) and Bob's observation number 7 (*all leather parts are very worn*). Figures 5 and 6 show that these two observations uses two *assessment* functions, f_2^A and f_5^B, and two *measurement* functions, f_1^A and f_4^B. It is obvious that, in Alice's and Bob's reasoning, the functions f_2^A and f_5^B plays the same *role*: to assess something about the system. Similarly, the role of f_1^A and f_4^B is to measure the level of some time function. It is then clear that the time functions $x_2^A(t)$ and $x_3^B(t)$, although basically different, play the same role in Alice's and Bob's reasoning. The same analysis holds for the others time functions. As a consequence, these two observations can be represented with the same pattern made of *type* of function linking type of *variable* (cf. figure 9). A set of such patterns is called the *functional network*. It is build from the projection from a *concrete conceptual space*, typically a blended space, to the space of the function's types. To build the figure 9, let us define the type of functions used by our three speakers.

The type of an *assessment* function is called *Discretization*: this is the function's type of the dynamic functions f_2^A, f_4^A, f_6^A, f_8^A, f_3^B, f_5^B and f_5^C. The *Discretization* function's type corresponds to the *Quantization* operation in the Discrete Event Systems community. It is represented with a function of the form $x_2^D = f_D(\Psi, x_1^D)$ where Ψ is a set $\Psi = \{\psi_i\}$ of *thresholds* values ψ_i. The definition domain of f_D is $\Delta_{x_1^D} \times \Delta_{x_2^D}$ where $\Delta_{x_1^D}$ is a *cardinal* set and $\Delta_{x_2^D}$ is an *ordinal* set or a set without any topology. The constants $\delta_i^{x_2^D}$ of $\Delta_{x_2^D}$ denote *ranges* of values (i.e. intervals) in $\Delta_{x_1^D}$ so that the number of elements in $\Delta_{x_2^D}$ is the numbers of thresholds values ψ_i in Ψ plus one. As a consequence, any function mapping a cardinal set to an ordinal set or an a-topology set can be represented with a f_D function type. In the running example, the time functions $x_2^A(t)$, $x_8^A(t)$, $x_6^A(t)$, $x_5^C(t)$, $x_4^A(t)$, $x_3^B(t)$ and $x_5^B(t)$ are linked with the variable's type x_2^D.

The type of an *assertion* function, the dynamic functions f_1^B, f_1^C, f_2^C and f_3^C, is a called *Classification*. A classification function implements a reasoning that uses a set R_f of classification rules of the form (N denotes the set of natural numbers):

$$\forall x_1^C \in \Delta_{x_1^C}, \exists n \in N, x_1^C = \delta_i^{x_1^C} \implies x_2^C = n \quad (3)$$

In this equation, n denotes a particular class so that $x_2^C = n$ means that the class corresponding to the value of x_1^C is the n^{th} class. A classification function is then represented with a function of the form $x_2^C = f_C(R_f, x_1^C)$, its definition domain being $\Delta_{x_1^C} \times N$ where $\Delta_{x_1^C}$ is any type of set. Any function mapping a set to N can be represented with a f_C function type. The time functions $x_2^C(t)$, $x_3^C(t)$, $x_1^B(t)$ and $x_1^C(t)$ are then linked with the variable's type x_2^C.

The type of a *measurement* function is a *Modeling* function. It concerns the dynamic functions f_1^A, f_5^A, f_7^A and f_4^B. It is represented with a function of the form $x_2^M = f_M(M_f, x_1^M)$ where M_f is a model providing the value of x_2^M given those of x_1^M. The definition domain of f_M is $\Delta_{x_1^M} \times \Delta_{x_2^M}$ where $\Delta_{x_1^M}$ and $\Delta_{x_2^M}$ are *cardinal* sets. Any function mapping two ordinal sets can be represented with a f_M function type. The time functions $x_1^A(t)$, $x_7^A(t)$, $x_5^A(t)$ and $x_4^B(t)$ are then linked with the variable's type x_2^M.

The type of a *counting* function is a *Numbering* function (f_3^A and f_4^C). A numbering function is a function of the form $x_2^N = f_N(P_f, x_1^N)$ where P_f is a counting process (i.e. a Poisson process or a discrete Markov counting model for examples). The definition domain of f_N is $\Delta_{x_1^N} \times \Delta_{x_2^N}$ where $\Delta_{x_1^N}$ and $\Delta_{x_2^N}$ are two *ordinal* sets: any function mapping two ordinal sets can be represented with a f_N function type. The time functions $x_4^C(t)$ and $x_3^A(t)$ are then linked with the variable's type x_2^N.

The last type of function of the running example corresponds to the *dating* function f_2^B: it is called the *Time-Stamping* function type and is represented with a function of the form $x_2^T = f_T(T_f, x_1^T)$ where T_f is a time stamping process providing the timestamps t_k of the current value of x_1^T. The definition domain of f_T is $\Delta_{x_1^T} \times \Delta_{x_2^T}$ where $\Delta_{x_1^T}$ can be any kind of set, $\Delta_{x_2^T}$ being an *ordinal* set. So, any function mapping a set to an ordinal set can be represented with a f_T function type. Only the time function $x_2^B(t)$ is linked with the variable's type x_2^N.

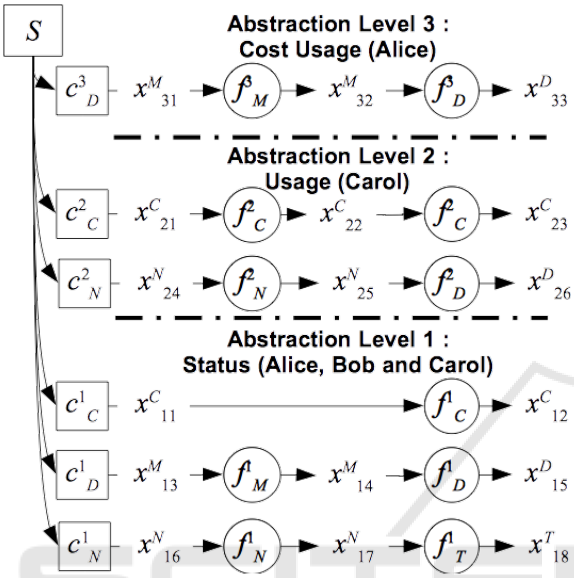

Figure 9: Generic Conceptual Space.

Mapping the dynamic and the time functions of the blended space of figure 8 with the corresponding types leads to the functional network of figure 9. The components have also been associated with *abstract components* so that C_1 is linked with the S component, C_3^A is linked with C_D^3, C_3^C is linked with C_N^2, C_2^A is linked with C_C^3, C_C^1 and C_D^1, the pair (C_3^B, C_5^B) is linked with C_1^N, C_4^B with C_D^1 and finally, C_2^C with C_C^1. An abstract component specifies the properties or the constraints that a concrete component must satisfy to be considered as an instance of this abstract component. The reference to any concrete element in the blend being contained in the projection from the blended to the generic conceptual space, the functional network is more compact than the blend.

An important remark is that, when forgetting the projection, there is no way to come back from the generic to the blended space. In other words, it is impossible to build Alice, Bob and Carol observations with the only functional network of figure 9: the blended space of figure 8 is a particular instantiation of the functional network of figure 9.

7 LEVELS OF ABSTRACTION

Fauconnier and Turner's theory aims at studying the *creation* of specific *structures* that *emerge* out of the blending operation. The level of abstraction of the emerging structures is then an inherent property of the structures themselves (cf. the three levels of abstraction in figures 8 and 9).

Nevertheless, Newell builds an *ontological* notion of level of abstraction where a level of abstraction describes a *system* that transforms a *medium* through its *components*, providing *primitive treatments*, and defines (Newell, 1981):

1. *laws of compositions* of the components to specify the contraints that any *structure* must satisfy at this level of abstraction, and
2. *laws of behavior* to establish how the system behavior emerges from a particular composition of its components.

Newell uses this notion to describe an information system with four levels of abstraction, the *physic* level (electromagnetic waves), the *circuits* level (transistors), the *logic* level (boolean algebra) and the *symbol* level (program), from the most concrete (continuous space) to the most abstract (purely discrete space), and proposes the existence of the *Knowledge Level* that it places above these ones. The Knowledge Level is characterized by the fact that there is no *law of composition* because the system behavior is governed by a *Principle of Rationality*: described at the Knowledge Level, a system is an *agent* whose components are *goals*, *actions* and a *body* (i.e. a knowledge corpus); and which processes its input informations to determine the (output) actions to take in order to reach its goals.

On an another hand, Floridi's uses an *epistemological* point of view to develop its *Method of Levels of Abstraction* (Floridi, 2008; **?**). A *level of abstraction (LoA) is a finite but non-empty set of observables* (Floridi, 2008, p. 10), and the word *system* refers to the object of study, a process in science or engineering or a domain of discourse. The *behaviour of a system, at a given LoA, is defined to consist of a predicate whose free variables are observables at that LoA. The substitutions of values for observables that make the predicate true are called the system behaviours*. A *Level of Abstraction* is then a particular organization of *variables*, *observables*, *behaviors* and *transition rules* between *values*. *A moderated LoA is defined to consist of a LoA together with a behaviour at that LoA*, (Floridi, 2008, p. 11).

The *Method of Levels of Abstraction* organizes LoA's in *Gradient of Abstraction* (GoA). A GoA allows to vary the LoA to make observations at different *granularity levels*: the higher the level of abstraction, the fewer but richer the information. *The quantity of information in a model varies with the LoA: a lower LoA, of greater resolution of finer granularity, produces a model that contains more information than a model produced at a higher, or more abstract, LoA*, (Floridi, 2008, p. 18). Foridi's theory distinguishes two kinds of GoA: *disjoint GoAs*, where the LoA are independent together, and *nested GoAs* where each LoA incrementally describes the same phenomena.

Tom4A defines three moderated LoA's. The most concrete is called the *Observation LoA*: the blended space of figure 8 constitutes the moderated LoA for the running example at the Observation Level of Abstraction. It formally describes the *observations* of the speakers according to the TOT mathematical framework. It is made with a set of *binary relations* linking concrete components, time functions and dynamic functions constituting respectively the *Structural Model*, the *Behavioral Model* and the *Functional Model* of the *observed process* $(X(t), \Theta(X,\Delta))$ (cf. section 3 and (Pomponio and Le Goc, 2014) for a detailed example).

The intermediate LoA is called the *Computing LoA*: the functional network of figure 9 is the moderated LoA for the example at the Computing Level of Abstraction. It formally describes the types of computing that are required to create *timed observations* from an observed process. The Computing LoA contains the necessary and sufficient corpus of knowledge to *specify* the programs that could generate the timed observations of the Observation LoA. It is made of *binary relations* linking *types* of components, variables and functions, describing the *logical* approach to build the timed observations at the Observation LoA.

The highest LoA according to Tom4A is the *Reasoning LoA*: it is made with at least one *inference structure* describing the way of using the type of functions of the Computing LoA to achieve a particular goal. It is made of *binary relations* linking *knowledge roles* and type of *inferences*, describing the elementary reasoning *steps* that are required to achieve a goal.

To build a model at this level of abstraction, let

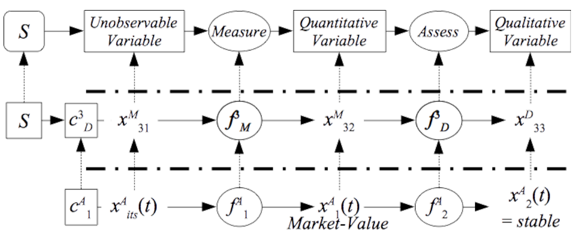

Figure 10: Alice's Observation 3 at three LoA.

us consider again Alice's observation *it has a stable market value*. The first point to notice is that Alice uses the term *market value* to build its observation. This term has been represented with the time function $x_2^A(t)$ (cf. figure 10), which is a *Discretization function* represented with the variable's type x_{33}^D. At the Knowledge Level, the role of a discretization function is to transform a *Quantitative Variable* in a *Qualitative Variable*. Such a transformation aims at defining the level of a quantitative variable regard to thresholds (cf. section 6). In the same spirit, the role of a modeling function f_M^3 is to provide a quantitative evaluation of the *Unobservable Variable* x_{31}^M which characterizes, at the Knowledge Level, the *phenomena* of the evolution of the time function $x_{its}^A(t)$. The role of *System* is then those of a *transfer function* graphically represented a rectangle with round corners (Schreiber et al., 2000): to provide values for each of its variables. Finally, according to Tom4A, the complete meaning of Alice's observation number 3 is given in figure 10.

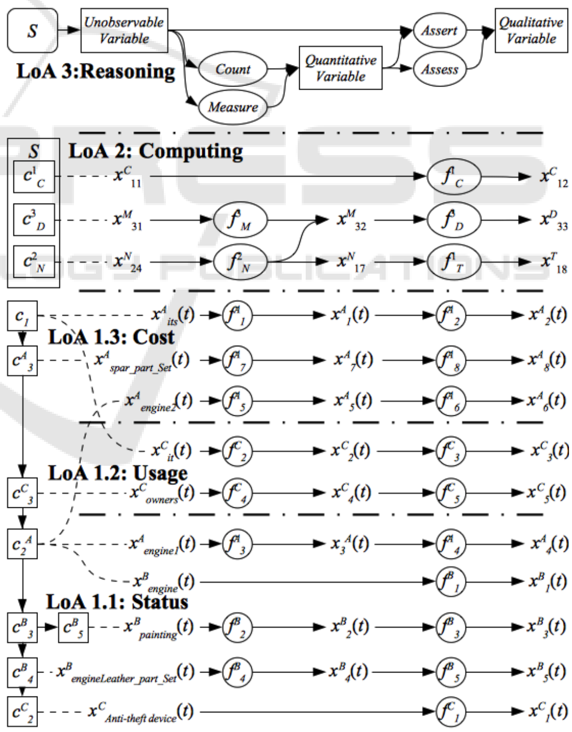

Figure 11: Gradients of Abstraction of the Conversation.

In figure 11, the blended, the generic and the inference conceptual spaces have been organized in two *Gradient of Abstraction* (GoA): a *disjoint GoA* which constitutes the lowest abstraction level, the *Observation LoA*, and contains the Blended Conceptual Space of figure 8, and a *nested GoA* made of the *Observation LoA*, the *Computing LoA* and the *Reasoning*

LoA. The Generic Conceptual Space of figure 9 is represented with the intermediate abstraction level, the *Computing LoA*. The effect of the conceptual equivalence appears clearly: even if they aim at representing the same thing, the functional network of figure 11 is much more compact than the Generic Conceptual Space of figure 9. Up to our knowledge, the *Reasoning LoA* has no counter part in the Blending Theory. The fundamental interest of this LoA appears in figure 11: it allows to identify the common aim of the speakers to explicit some properties of a (still unknown) system in order to state its qualities and defects. A concrete illustration of such a disjoint and nested GoA can be found in (Le Goc, 2004).

8 CONCLUSION

This paper proposes a formal framework, called Tom4A (Timed Observations Method for Abstraction), that provides for the first time, up to our knowledge, a strong mathematical foundation to both the Blending Theory (Fauconnier and Turner, 1998) and the Method of Abstraction Theory (Floridi, 2008). Constructed on the *Timed Observations Theory* (TOT), Tom4A completes the Tom4D Knowledge Engineering methodology (Timed Observations Methodology for Diagnosis, (Pomponio and Le Goc, 2014)) and the Tom4L Knowledge Discovery in Databases process (Timed Observations Mining for Learning, (Le Goc et al., 2015; **?**)), also based on the TOT. The basic concepts of Tom4A are progressively introduced with a running example, an exchange between three speakers, whose original text comes from the web site of the *Society for the Philosophy of Information* (http://www.socphilinfo.org/node/150). This example provides for the first time, still up to our knowledge, the first conceptual model of such an exchange under the formal form of two gradients of abstraction, defining the meaning of this exchange.

Our long term goal is to develop software tools able to discover and to model knowledge representations from sets of timed data so that the human interpretation is intuitive, immediate and independent of the learning and the modeling tools. The next step of this work is then to propose a new formalization of the analogical reasoning based on the combination of the TOT and the Category Theory (Mac Lane, 1978).

REFERENCES

Bredeweg, B. (1994). *The CommonKads Library for Expertise Modelling, chapter Model-based diagnosis and prediction, p.121-153*. IOPress.

Fauconnier, G. and Turner, M. (1998). Conceptual integration networks. *Cognitive Science*, (22):133–187.

Fauconnier, G. and Turner, M. (2003). Conceptual blending, form and meaning. *Recherches en communication, n° 19 (2003).*, (19):57–86.

Floridi, L. (2008). The method of levels of abstraction. *Minds and Machines*, 18:303–329.

Floridi, L. (2010). Levels of abstraction and the turing test. *Keybenetes*, 39(3):423–440.

Le Goc, M. (2004). Sachem. a real time intelligent diagnosis system based on the discrete event paradigm. *Simulation, The Society for Modeling and Simulation International Ed.*, 80(11):591–617.

Le Goc, M. (2006). *Notion d'observation pour le diagnostic des processus dynamiques: Application à Sachem et à la découverte de connaissances temporelles*. Hdr, Aix-Marseille University, Faculté des Sciences et Techniques de Saint Jérôme.

Le Goc, M. and Ahdab, A. (2012). *Learning Bayesian Networks From Timed Observations*. LAP LAMBERT Academic Publishing GmbH & Co. KG.

Le Goc, M., Barthelot, F., and Pascual, E. (2015). Emergence of regularities in the stochastic behavior of human. In *IEEE International Conference on Data Mining Workshop, ICDMW 2015, Atlantic City, NJ, USA, November 14-17, 2015*, pages 381–388. IEEE Computer Society.

Le Goc, M. and Gaeta, M. (2004). Modeling Strutures in Generic Space, a Condition for Adaptiveness of Monitoring Cognitive Agent. *Journal of Intelligent and Robotics Systems*, 41(2-3):113–140.

Mac Lane, S. (1978). *Categories for the Working Mathematician*, volume 5. Springer-Verlag New York.

Newell, A. (1981). The knowledge level. *AI Magazine*, 2(2):1–20.

Pomponio, L. and Le Goc, M. (2014). Reducing the gap between experts' knowledge and data: The tom4d methodology. *Data & Knowledge Engineering*, DOI 10.1016/j.datak.2014.07.006.

Schreiber, G., Akkermans, H., Anjewierden, A., de Hoog, R., Shadbolt, N., de Velde, W. V., and Wielinga, B. (2000). *Knowledge Engineering and Management: The CommonKADS Methodology*. MIT Press.

Zanni, C., Le Goc, M., and Frydman, C. (2006). Conceptual equivalence at knowledge level for diagnosis applications. *KES, International Journal of Knowledge-Based and Intelligent Engineering Systems*, 10(3):225–246.