Generative Ontology of Vaiśeṣika

Rajesh Tavva[1] and Navjyoti Singh[2]

[1,2]Center for Exact Humanities, International Institute of Information Technology, Hyderabad, Telangana, India
[1]vrktavva@research.iiit.ac.in and [2]navjyoti@iiit.ac.in

Abstract

In this paper we present a foundational as well as generative ontology which is graph-based. We use a form called *punctuator* which is non-propositional and also non-set-theoretic to build our system – *Neo-Vaiśeṣika Formal Ontology*. The idea is to present an ontological language which is formal. This language is a set of potentially infinite sentences (graphs) whose structure is captured by a finite set of (graph) grammar rules. We also have an interpreter to interpret the graphs generated by this grammar and show that the interpretation of a node as belonging to a particular ontological category is based purely on its structure/form and nothing else to provide a robust example of a formal, foundational and generative ontology.

Keywords. Formal Ontology, Graph Grammar, Generative Ontology, Punctuator, Generative Grammar, Vaiśeṣika

Introduction

Most, if not all, of the (computational) ontologies built till now [12] are either built manually or through automatic methods of category-extraction from text. There is no notion of generation there since there are no repeating structures (each category is different and hence different structure). All the foundational ontologies are presented as diagrams/graphs with finite number of nodes (which stand for categories/sub-categories) and edges (which stand for class-subclass or some other relations). In this paper we present a novel concept of *Generative Ontology* which presumes *Grammar of Reality* which, in turn, is based on the idea of a recursive ontological form called the punctuator. A punctuator is a form which is present between any two entities and enables us to distinguish one from the other. It is because of this form that we are able to differentiate various categories as well as different instances of the same category. Since a punctuator is found not only between two classes, but also between two particulars, or between a class and a particular, the graphs which are based on punctuators contain not only classes/universals, but also particulars. But since the punctuator has a recursive form, complex graphs can be generated from simple graphs, and each of these graphs depicts some or other portion of reality at some or other level of granularity. Hence we also have an interpreter to interpret the generated graphs as portions of reality.

We take Vaiśeṣika - one of the Indian philosophical schools which focuses on foundational ontology – as formalized in [4,5,6,7] as our base, and present generative as well as interpretative grammars for it in this paper. This paper's focus is not on defending Vaiśeṣika description of reality or the rationale behind its categorial system[1]. This paper is already taking them as given and trying to make implicit formal notions of Vaiśeṣika explicit. The idea is to show the possibility of an ontological language which can be formalized and also generated, and then interpreted.

In section 1 we present Vaiśeṣika ontology in brief. In section 2 we present Neo-Vaiśeṣika Formal Ontology, the system we built by formalizing Vaiśeṣika ontology. In this section, we give formal definitions of Vaiśeṣika categories in terms of three basic punctuators. In section 3 we show how Vaiśeṣika can be seen as a generative ontology. In section 3.1, we give the generative grammar of our system where we give the production rules to generate graphs, and in section 3.2 we give interpretative rules using which we can interpret these generated graphs to label each node with some or other Vaiśeṣika category. The generated graph is considered to be valid if there is at least one interpretation, in terms of Vaiśeṣika categories, of the generated graph. The grammar is considered to be sound if it generates only valid Vaiśeṣika graphs. Though a formal proof of the soundness of this grammar is not achieved yet, we present in section 4 some results which show that the system is promising. The main focus of the paper is to present the idea of a generative ontology and one example of such an ontology.

We tried to provide examples and explanation of abstract concepts wherever possible but due to space constraints we are unable to get into detailed explanation sometimes. An extended version of this paper with examples, diagrams and detailed explanation of various

[1] One can refer to [4] and [14] for that.

definitions, axioms etc. presented in this paper is available at [9].

1. Vaiśeṣika Ontology

Vaiśeṣika, as mentioned in introduction, is one of the many Indian philosophical schools, which focuses on foundational ontology. It classifies all entities of reality into 6 categories[2]: (1) Substance (e.g: material entities like tables, chairs as well as non-material entities like soul, space and time) (2) Quality (e.g: color, weight) (3) Action (e.g: rising up, falling down, motion) (4) Universal (e.g: tableness, chairness, redness) (5) Ultimate Differentiator (located in each ultimate substance (explained below) and differentiates one from the other) and (6) Inherence (explained below). Notice that there is no category like *Relation* in this system because any chain constituting a *relational context* (section 2) can be abstracted into something called a *relation*. Out of all the categories listed above *inherence* (Fig. 2.2) is the most significant one for us. It gives stability to reality and is the entity constituting the second most pervasive relation found in reality, the first being *self-linking relation* (more about it later). We call the relational chain constituted of inherence entity as inherence relation, and this inherence relation can be found in all the following cases: (1) Universals inhere in substances, qualities and actions (2) Qualities and actions inhere in substances (3) Substances (wholes) inhere in other substances (their parts) and (4) Ultimate Differentiators inhere in ultimate substances. Ultimate substances (US) are those substances which have no parts and hence cannot be differentiated from each other[3]. Examples of USs are time, space, souls and atoms (these are different from those of physics, they are the smallest indivisible units according to Vaiśeṣika). Since USs have no parts, each of them has an ultimate differentiator (UD) inhering in it which differentiates it from the rest of the entities. The remaining substances which are not USs are automatically differentiated from others since they are wholes/mixtures/compounds made up of various USs which are already differentiated by UDs. We will try to define all these categories *formally* using only the idea of inherence.

2. Neo-Vaiśeṣika Formal Ontology

Vaiśeṣika ontology, due to Kanada [1], Prasastapada [2] and Udayana [3] has been formalized by Navjyoti [4,5,6,7] which resulted in *Neo-Vaiśeṣika Formal Ontology.* From now we will refer to this formal system as *F* for short. We explicate the system below.

According to *F* reality is constituted of only entities and punctuators. Here entities refer to all kinds of beings like objects, events, relations and so on. And any two beings/entities are considered two, not one, because there is a separator/boundary/vacuum/non-being between them which we call *punctuator/point*[4]. But a punctuator is not only a separator but also a connector/link which links two entities. So here non-being is not an all-encompassing vacuum whose existence Parmenides[5] was denying but every non-being is a *particular* in the sense that it rides on a particular pair of entities/beings which it separates as well as brings together into contiguity[6] in some relational context. Our idea of punctuator/point is quite similar to, at the same time slightly different from, Leibniz's idea of punctum/point and Brentano's idea of punctiform/point. Leibniz [10] tries to explicate this notion by taking a geometrical example like a line segment. He makes a distinction between potential point (euclidean point) and actual point (non-euclidean point). The line segment, according to Leibniz, is not really made up of euclidean points, they are not parts of the segments the way smaller segments are. The euclidean points (lengthless, breadthless and heightless points) are only potentially there on the segment but not actually constituting it. The actual points are the boundaries or the endpoints of the segment. These, Leibniz also calls as punctums because they punctuate the line segment from its neighbourhood (in one-dimensional space). A potential (euclidean) point can become an actual (non-euclidean) point if one cuts the line segment, say, into two halves. The midpoint which was there before cutting was only potential, but after cutting it becomes actual. So according to Leibniz a punctum is like a boundary that separates two entities. Brentano [11] also comes up with a similar idea and he calls it punctiform or point. He says no continuum can be built up by adding one individual point to another. The point exists as a boundary, as a limit. Similarly for us, a point is a boundary between two entities, but not constituting those entities.

These both – *entity* and *punctuator* – are defined in *F* in terms of each other. But the mutual dependency of definitions ends here. All the other definitions are based on these primitives only.

Definition 2.1: An *entity* is that which is never punctuated from itself.

Definition 2.2: A *punctuator* $p_r(x|y)$ has the form $<x|y, R>$ where x and y are entities punctuated from each other in some relational context R. This R refers to all the chains of entities (along with punctuators between them) connecting x and y where each chain looks like $C = p(x|e_1) – p(e_1|e_2)... - p(e_i|e_{i+1}) - ...p(e_{n-1}|e_n) – p(e_n|y)$ where each e_i is

[2] A detailed list of Vaiśeṣika categories and subcategories can be found in a tabular format in [13].

[3] This notion of parthood is quite different from other mereological notions like that of [15]. For detailed explanation of the notion of parthood in Vaiśeṣika please refer to [4].

[4] If the separator were also a being, then one would run into infinite regress. Hence one needs to accept a non-being like punctuator.

[5] http://plato.stanford.edu/entries/parmenides/

[6] Two entities in contiguity are related to and yet distinct from each other.

also an entity.

Notice that the punctuator is neither the entities nor the relation between them. It is the form/arrangement of all these things put together. Though an entity is existentially independent of its punctuations with other entities, its meaning solely comes from these punctuations. Hence the form of an entity defines its meaning. In this sense *F* satisfies Husserl's definition of formal ontology: *"eidetic science of the object as such."*[7]

2.1. Three Basic Punctuators and Abstract of Punctuator

From our knowledge of Vaiśeṣika we have narrowed down to three basic punctuators/forms in *F*. It seems that these three basic forms are sufficient to derive any other complex form in the universe. Their definitions and other details are given below.

Definition 2.3 (Self-Linking Punctuator): A *self-linking punctuator* is one whose relational context is empty. It is represented as $p_{sl}(x|y)$ and its structure is $<x|y, <>>$.

A self-linking relation is one in which one of the relata itself acts as the relation. The relata don't need a third entity to bring them together. The relation is called self-linking not because it links with itself but because it links by itself to other entities. The punctuator corresponding to the self-linking relation is called the self-linking punctuator. It can be represented as in Fig. 2.1.

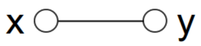


Fig. 2.1. *Self-linking punctuator*

Here the edge being undirected only means that the direction is changeable (between the same pair of entities) in the case of self-linking punctuator. But in the case of inseparable punctuator that is not the case.

Definition 2.4 (Inseparable Punctuator or Inherence punctuator): An *inseparable punctuator* is one which has an inert entity called *I* (inherence) which inseparably binds two entities *x* and *y* . It is represented as $p_{in}(x|y)$ and it has the structure $<x|y, p_{sl}(x|I) - p_{sl}(I|y)>$.

This *I* is nothing but the inherence category of Vaiśeṣika discussed above. It can be represented as in Fig. 2.2. Here the chain connecting *x* and *y* via *I* can be abstracted into a relation called inherence relation, and the entity *I* can be said to be playing a relational role only when it is part of such a chain. This chain, along with both *x* and *y* constitutes the inherence punctuator between *x* and *y*. Hence a relation and punctuator are different.

Unlike self-linking relation where one of the relata itself is playing the role of the relation here a third entity, a relational entity, *I* (inherence) is necessary to bring two entities *x* and *y* into inherence relation. It is called inseparable because inherence relation itself cannot be destroyed without one of its relata being destroyed. Various cases where inherence relation is found in reality are given in section 1.

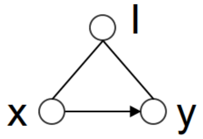


Fig. 2.2. *Inseparable punctuator*

Inherence relation repeats almost everywhere in the universe. Hence it is one of our most fundamental relations to derive other complex relations. In fact if we take all the instances of inherence relation in the universe we will get the entire synchronic reality or the snapshot of the reality. To get diachronic reality, we need to look at the changes which occur in it for which we need to move to the next punctuator.

Definition 2.5 (Separable Punctuator): A *separable punctuator* $p_{cd}(x|y)$ with structure $<x|y, p_{sl}(x|I) - p_{sl}(I|X) - p_{sl}(X|I) - p_{sl}(I|y)>$ where *X* is either of the relational entities *C* called conjunct (or contact) or *D* called disjunct. *C* (or *D*) binds *x* and *y* in such a way that they are associated (or dissociated). $p_{cd}(x|y)$ switches between two structures that can be represented in short as $<x|y, w_C>$ and $<x|y, w_D>$ since no two entities can be in both conjunct and disjunct at the same time. It is shown in Fig. 2.3.

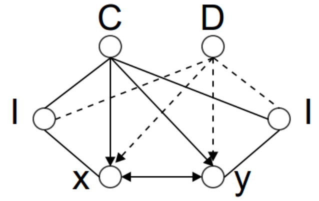


Fig. 2.3. *Separable punctuator*

The relation between *C* (or *D*) and its relata is inherence. If two entities *x* and *y* are in conjunct (or disjunct) then the conjunct (or disjunct) with *y* inheres in *x* and vice-versa. The conjunct and disjunct which we are referring to here are completely different from the logical operations – conjunction and disjunction – which are used in propositional logic. They only refer to the relation of two entities being in contact (as in when we clap both our hands are in contact) or not in contact.

One can notice that the 2nd and 3rd punctuators are recursive in nature in the sense that the separable punctuator is made of two inherence/inseparable punctuators while the inherence/ inseparable punctuator is made of two self-linking punctuators put together. In this

[7] Edmund Husserl, Ideen zu einer reinen Phänomenologie §10, Husserliana 3/1: 26–7

sense, self-linking punctuator is the most fundamental of these 3 punctuators, but we will use all three of them as building blocks of our system since their patterns repeat everywhere and each of them plays a role in deriving more complex punctuators.

We need to look at some more definitions and axioms before we proceed further.

Definition 2.6 (Abstract of punctuator): *abs(p)* maps relational context of punctuator to two abstract entities ${}_{a}e$ and ${}^{a}e$ which are attributes of the two punctuated entities e_i and e_k respectively.

$$abs(p(e_i\,|e_k)) \rightarrow {}_{a}e \in F : p_{sl}\,(e_i \mid {}_{a}e)\ \&\ {}^{a}e \in F : p_{sl}\,(e_k \mid {}^{a}e)$$

To take an example, if color (e_i) inheres in table (e_k), it can also said to be located in table. So color is the locatee (${}^{a}e$) whereas table is the locus (${}_{a}e$). So the abstract of the inherence punctuator between color and table gives rise to two abstract entities, namely locus and locatee where locus, the attribute of table, is related to it by self-linking punctuator, and locatee, the attribute of color, is related to it by self-linking punctuator. Similarly there can be various other abstracts of punctuators like predecessor-successor, qualifier-qualificand, expression-expressed etc. which can be abstracted directly from one of the three basic punctuators or from some complex punctuator defined using them. Collection of pairs of each kind forms a contiguum like locational contiguum, succession contiguum etc.

2.2. Four Contigua

According to us there are four important contigua to be studied formally, to cover a major portion of reality, in the sequence listed below:

Locational contiguum: This is constituted of all the locus-located pairs of abstract entities. Study of this gives us entire synchronic reality.

Succession contiguum: This is constituted of all the predecessor-successor pairs of abstract entities. Study of this, along with locational contiguum, gives us entire diachronic reality.

Qualification contiguum: This is constituted of all the qualifier-qualificand pairs of abstract entities. The qualifier-qualificand relation is usually found in cognition. For instance when we see a white cloth, the structure of our cognition is of the form - qualifier (white) and qualificand (cloth) - whereas the form of entities in reality as such is that of locus (cloth) and located (white). Study of this contiguum, along with the previous two, is the study of how world is cognized, and the notions of knowledge, reasoning, truth etc. come at this stage.

Expression contiguum: This is constituted of all the expression-expressed pairs of abstract entities. Study of this contiguum, along with the previous three, is the study of the relation between language and reality. All language related issues like synonymy, polysemy etc. come at this stage.

We think that the four contigua listed above need to be studied formally in the specified order since each contiguum is dependent, for its study, on its previous one. This is a long-term project and this paper can be considered as the first step in that direction. Here we are doing a formal study of only the first contiguum and that too a portion of it. Formal study of the remaining portion of it, as well as formal study of other contigua is part of the future work.

Now we are all set to define the fundamental categories of Vaiśeṣika formally in our system *F.*

2.3. Fundamental Categories of Vaiśeṣika Defined Formally in *F* [8]

We will formally define the fundamental categories of Vaiśeṣika in *F* by doing a functional study of the locational contiguum of inherence punctuator for which we should first define something called the *Inherence Bifunction*.

Definition 2.8 (Inherence bifunction): The *Inherence Bifunction*, $IB(e) = ({}_{e}L,\ {}^{e}L)$ takes an entity *e* and returns the locational ranges of *e.*

It is called a bifunction because it works on two different things at the same time – locatee range and locus range of *e*. Locatee range of *e* refers to all the entities which are located in *e* (by inherence) and locus range of *e* refers to all the entities in which *e* is located (by inherence).

Now we will define categories by noticing some invariances in the output.

Definition 2.9: If $IB(e) = ({}_{e}L,\ Ø)$ then *e* is an *Ultimate Substance* (US) where Ø stands for the empty set.

So if there is an entity which inheres nowhere but has some other entities inhering in it, then it is said to be an ultimate substance (US). In other words US is an unlocated locus. Some examples are space, time, soul, atoms. (These are not yet defined in the system, but they are mentioned just to give more clarity on the nature of US).

Definition 2.10: If $IB(e) = (Ø, 1_{US})$ then *e* is an *Ultimate Differentiator* (UD) where 1_{US} stands for the unique structure $<e|US,\ p_{sl}\,(e|I) - p_{sl}\,(I|US\,)>$.

A UD is one in which nothing is located, but it itself is located in one and only one US. The role of UD is to differentiate one US from another since the USs are partless and cannot differentiate themselves from each other.

Axiom 2.1: Any US cannot have more than one UD located in it because a single UD in a locus is sufficient to distinguish its locus from the rest of the entities, and the second UD only becomes redundant.

Definition 2.11: If $IB(e) = (Ø,\ 2+)$ then *e* is a *Universal* (U).

[8] Detailed explication of these definitions along with diagrams and examples can be found in [9].

This says that a universal (U) is something in which nothing inheres, but it itself inheres in 2 or more (hence *2+*) loci i.e. its instances. A universal can inhere not only in US but also in non-US, hence *2+* is not subscripted with US.

Axiom 2.2: Given any two universals only one of the following two relations is possible between them: (1) Both of them are mutually exclusive i.e. both of them have no instances in common or (2) One of them subsumes the other i.e. all the instances of one universal are also the instances of the second universal whereas the second universal has some more instances which the first doesn't have. This gives rise to subclass-superclass structure among universals.

Definition 2.12: *If IB(e) = (1+, 1)* then *e* is a *Quality* (Q).

This says that a quality is that entity in which one or more entities inhere, and it itself inheres in only one entity.

As of now we are not making a distinction between Vaiśeṣika categories – Quality and Action – in our formal system *F*. That distinction is part of our future work, and for the time being we will refer to both as Qualities (Q).

Now we'll further analyze US for which we need to define the following function.

Definition 2.13: The function, *Locality of any Ultimate Substance* is defined as $LUS(e_u) = (J_C, J_D)$ where J_C is the set of structures conjoint with e_u and J_D is the set of structures disjoint with e_u.

Given e_u (a US) as input, the above function returns the ordered pair – the set of all entities in conjunct with e_u, and the set of all entities which are in disjunct with e_u. We will again look for invariances in the output to further analyze US category.

Definition 2.14: If $LUS(e_u) = (J_C, Ø)$ or if $LUS(e_u) = (Ø, J_D)$ then e_u is called a *Ubiquitous Ultimate Substance* (UUS).

This says that those ultimate substances which are only in conjunct (and not disjunct) relation with other entities or only in disjunct (and not conjunct) relation with other entities are defined as ubiquitous ultimate substances.

Definition 2.15: Those ultimate substances which are not ubiquitous ultimate substances are called *Mobile Ultimate Substances* (MUS) i.e. if e_u is an MUS then $LUS(e_u) = (J_C, J_D)$ where neither J_C nor J_D is Ø i.e. they are in contact with some while in disjunct with some others.

Axiom 2.3: UUSs are in contact only with MUSs, not among themselves[9].

Now we get back to inherence bifunction to define one last category of Vaiśeṣika i.e. Substantial Whole (SW).

Definition 2.16: If $IB(e) = (1+, 2+_{MUS})$ then *e* is a *Substantial Whole* (SW).

This says that an SW is that in which one or more entities inhere, and it itself inheres in two or more MUSs. These MUSs are its parts. For instance a table inheres in all its parts and hence is present in each of its part. That is the reason why the entire table moves even when a part of it is moved, or the table as a whole is cognized even when a part of it is perceived. And what inheres in a table can be its qualities or universals like tableness, substanceness etc.

Axiom 2.4: Given two parts of a whole there has to be at least one chain of contacts connecting them directly or indirectly.

Axiom 2.5: Two SWs are in contact with each other if one or more parts of one SW are in contact with one or more parts of the other SW.

Axiom 2.6: Two different SWs cannot have overlapping parts i.e. there can be no MUS in which two different SWs can inhere.

Axiom 2.7: An SW cannot have qualities inhering in it if none of its parts has quality/qualities inhering in it.

For instance a cloth cannot be white if none of its threads has whiteness in it.

Given the above definitions let's have a look at a sample valid graph of *F*, say, that of a substantial whole. It is depicted in Fig. 2.4.

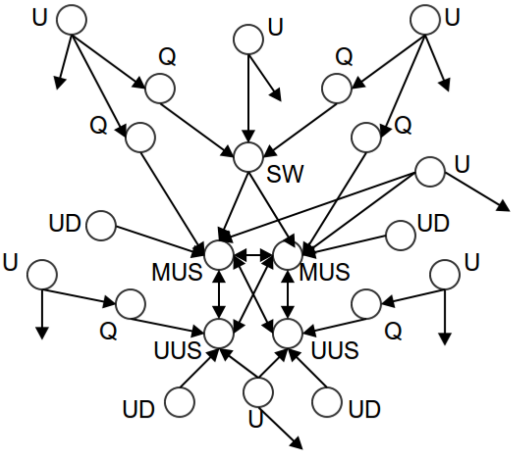


Fig. 2.4. *Ontology of a Substantial Whole*

It's easy to understand this graph if we start with the focal point of it which is a substantial whole (SW). This SW is inhering in two mobile ultimate substances (MUS) (it can inhere in more than two MUSs as well) which are in contact with each other[10]. Now there are also two UUSs which are in contact with each of the MUSs. Each of the ultimate substances – two MUSs and two UUSs – has an ultimate differentiator (UD) inhering in it. Each of the substances – SW, MUS and UUS – has one quality (more than one is also possible) inhering in it, and also one universal (more than one is also possible) in it. Each of the qualities has one universal (more than one is also possible) in it. The universals are shown with dangling edges, it is to

[9] Refer to [4] for the rationale behind this axiom.

[10] Each bidirectional edge here is a short form for the separable punctuator, in contact position. The contact entities are currently hidden for aesthetic purposes. Similarly the unidirectional edges stand for inherence punctuators and the inherence entities are hidden as well.

show that the universals (can) inhere in more entities which are not shown in this graph. Notice that no UD has a universal inhering in it because it would violate its definition (Definition 2.10).

We can take a particular substantial whole, say, a *table* to exemplify the above graph. In this example each of the above categories stand for the following:

- SW – table
- MUSs – atoms of table
- UUSs – space and time
- Universals inhering in table – tableness, substanceness
- Universals inhering in MUSs – mobile-ultimate-substanceness, substanceness
- Qualities inhering in table – color, size
- Universals inhering in these qualities – colorness, sizeness, qualityness
- Qualities inhering in atoms of table – color, touch
- Universals inhering in these qualities – colorness, touchness, qualityness
- Universals inhering in UUSs – ubiquitous-ultimate-substanceness, substanceness
- Qualities inhering in ultimate substances – size, number
- Universals inhering in these qualities – sizeness, numberness, qualityness

The no. of entities (Qs, Us, MUSs etc.) presented in this example don't match exactly with those presented in Fig. 2.4. The above figure is only a kind of template to understand this example.

In the next section we present generative rules - to generate the structures as in Fig. 2.4 – as well as interpretative rules - to label the nodes in these structures with the categories of Vaiśeṣika.

3. Vaiśeṣika as a Generative Ontology

A graph grammar is a generalization of string grammars and tree grammars. The Left-Hand-Side (LHS) and Right-Hand-Side (RHS) of a production/transformation rule in graph grammar are both graphs instead of strings or trees. The rules modify a host graph into a different graph by replacing a subgraph of it which is matching with LHS, with an incoming graph - RHS. The below production/transformation rules will be self-explanatory except, may be, for Negative Application Condition (NAC). It is defined below.

Definition 3.1: Rules can have exceptions - it may not be likely to apply a rule in some particular cases. Those cases/conditions are called Negative Application Conditions (NACs). NACs can also be depicted in the form of graphs. Hence each rule may (or may not) have one or more NACs. So a rule will be applied only when its LHS matches with some subgraph of the host graph, and none of its NACs matches with any subgraph of the host graph.

For a detailed introduction to graph grammars please refer to [8].

3.1. Generative Rules of Graph Grammar of Vaiśeṣika

Any generative grammar will have a start symbol, some alphabet and some production rules involving the symbols of the alphabet. Similarly our grammar will also have a start graph, an alphabet which consists of a set of node-labels (Ω_V) and a set of edge-labels (Ω_E), and production rules to generate various graphical structures from these symbols.

One can treat the generative rules of Vaiśeṣika as the syntactic portion and its interpretative rules as the semantic portion. The generative rules generate the graphs as pure symbolic structures without any meaning as such whereas the interpretative rules add meaning to these symbolic structures by labeling each node with some category of Vaiśeṣika.

In our grammar, the node-labels are $\Omega_V = \{g, C, D, h, i, p, q, r, s, u, v, e\}$ and edge-labels are $\Omega_E = \{sl, in, con, dis\}$. Each of the node-labels stands for the following: *g* – start node (this is the only node in the start graph), *C* – conjunct entity, *D* – disjunct entity, *h*, *i*, *p*, *q*, *r*, *s*, *u*, *v* – are all various node labels used in the process of generation. At the end of the generation all of them will be replaced by a common label, *e*, to show that the nodes they were labeling can be interpreted later based purely on their structures and not on their labels. And the edge labels stand for the following: *sl* – self-linking relation, *in* – inherence relation, *con* – conjunct relation, *dis* – disjunct relation. But in the rules below we have differentiated edges based on their arrows instead of their labels for aesthetic purposes. A self-linking relation has no arrows (though it's asymmetric its direction is changeable), an inherence relation has one arrow (it's asymmetric and its direction is fixed), conjunct relation has two arrows and a thick line whereas disjunct relation has two arrows and a dashed line (both are symmetric relations).

No two entities have more than one edge (of any type) between them. That is a default NAC for every rule and hence not being specified with each rule.

Currently the rules are generated keeping Vaiśeṣika categories and their corresponding invariant structures in mind. Intuitively they seem to generate valid Vaiśeṣika graphs but the proof of soundness is necessary to prove it formally. That is not achieved yet and is one of our future goals.

We give below the rules as well as the rationale in coming up with them. If we consider USs to be the bottom of universe (since they inhere nowhere else i.e. not located anywhere) and universals and UDs to be the top of universe (since nothing inheres in them i.e. nothing is located in them), then we are trying to generate all the graphs from bottom to top. The rules are prioritized by dividing them into layers - once you are in n^{th} layer, you cannot apply any rules from layer 1 to n-1. But if there are multiple rules in a given layer they can be applied in any order.

We start with the first layer. It has only one rule in which we replace the start node *g* with two other nodes – *h* and *i*.

LHS	RHS
○g	○ h ○ i

Here *h* is intended to generate mobile ultimate substances whereas *i* is intended to generate ubiquitous ultimate substances. Those rules follow in coming layers.

The second layer also has two rules. In the first rule, *h* is replaced by itself and another node *p*. This is intended to generate as many *p*s as one wants and in the second rule of this layer, *h* is replaced just with *p*. This is to terminate the process of generation of *p*s.

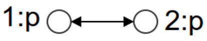
We define the graphs generated in a particular layer to be the terminal graphs (TGs) of that layer if no more graphs can be generated from them using the rules of that layer, otherwise we will call them non-terminal graphs (NTGs).

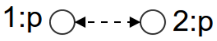
LHS	RHS
○ h	○ h ○ p

LHS	RHS
○ h	○ p

These *p*s are supposed to stand for mobile ultimate substances. Once the generation process in this 2nd layer terminates i.e. no more *h*-labeled nodes remain, the remaining graphs are TGs of layer 2, and it is on these graphs that we will apply the rules from layer 3.

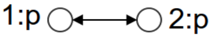
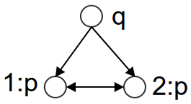
In the 3rd layer we have two rules – one to create contacts among *p*s and another to create disjuncts among them.

LHS	RHS
1:p ○ ○ 2:p	

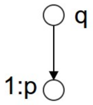
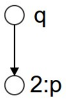
LHS	RHS
1:p ○ ○ 2:p	

Here the numbers '1:' and '2:' are used to map a particular node of LHS with a particular node of RHS (and NAC, if there is one). In our system this mapping is injective i.e. no two nodes in LHS can map to the same node in RHS or NAC and vice-versa. Once we get all the TGs of this layer, we will move to 4th layer.

Now in the 4th layer we will have rules for generating wholes, represented by *q*-labeled nodes. It has 2 rules, we will apply both these rules only on the TGs of 3rd layer, not others. In the first rule we state that if two *p*s (MUSs) are in contact, then let a new entity *q* (whole) inhere in both of them.

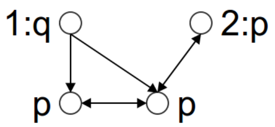
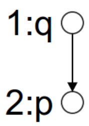
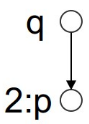
LHS	RHS
	

This rule has 2 NACs. These NACs state that the *p*s in which *q* is inhering shouldn't already have a whole (*q*) inhering in them. They are depicted below.

NAC_1	NAC_2
	

Notice that numbers like '1' and '2' are prefixed only to *p*-labeled-nodes and not *q*-labeled-nodes. It's because there is no such node in LHS to map with the nodes in RHS or in NACs. So the rule, along with its NACs, states that a new whole (*q in RHS)* inheres in two MUSs (*p*s) in contact if and only if there is no whole (*q in NACs*) already inhering in one or both of them. These NACs are necessary for us because, in our system, no two substantial wholes can inhere in the same part (Axiom 2.6). So even if one of the above NACs is found in that portion of the graph which matched with LHS, then this rule will not be applied.

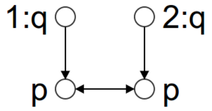
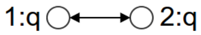
The above rule will produce only wholes inhering in two entities. But if we need wholes inhering in more than 2 entities upto an indefinite number, may be covering all *p*s (MUSs) in a step-by-step manner we need the following rule.

LHS	RHS	NAC
		

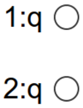
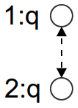
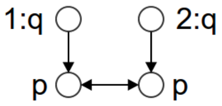
This rule has one NAC which states that no whole (*q*) should already inhere in 2:*p*.

Notice here that the *q*-node in NAC is not numbered whereas the *q*-node in RHS of the rule is numbered. The RHS is saying that an inherence edge should be added from the *same q* as in LHS while NAC is referring to *any q* inhering in *p*, not necessarily the same one as in LHS.

The next layer i.e. 5th layer has two rules. These can be applied on both TGs as well as NTGs of layer 4 since the latter (NTGs of layer 4) are also considered as valid intermediate graphs for the process of generation. The first rule states that if two MUSs (*p*s) are in contact then the wholes (*q*s) inhering in each of them can also be in contact. This is to model the fact that the contact between wholes can be inferred from the contact between their parts (Axiom 2.5).

LHS	RHS
	

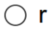
The 2nd rule of this layer is to handle disjuncts among wholes. This is like the complement of the first rule. This says that two wholes can be in disjunct if none of their parts are in contact. So this negative condition becomes the NAC of this rule.

LHS	RHS	NAC
		

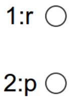
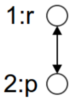
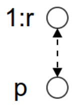
Till now we had rules which generate mobile ultimate substances, contacts or disjuncts among them, wholes inhering in mobile ultimate substances, and contacts or disjuncts among them. In the next couple of layers we introduce rules to generate ubiquitous ultimate substances and contacts or disjuncts - between them and other entities (since ubiquitous ultimate substances are in contact or disjunct only with mobile ultimate substances, not among themselves (Axiom 2.3)).

The next layer i.e. 6th layer has two rules. They apply only on the TGs of 5th layer. The *i*-node which we generated in our 1st layer is now used to generate ubiquitous ultimate substances represented by *r*-labeled-nodes or *r*-nodes. The two rules here are exactly same as the ones in 2nd layer, only with labels differing in both LHS and RHS.

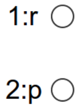
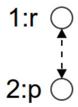
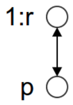
LHS	RHS

LHS	RHS
	

In the 7th layer, we have two rules. They apply on the TGs of 6th layer. The first rule states that an ubiquitous ultimate substance can be in contact with a mobile ultimate substance, if it is not in disjunct with any. So it has one NAC to specify this negative condition.

LHS	RHS	NAC
		

The second rule is very similar to the first one. It says an ubiquitous ultimate substance can be in disjunct with a mobile ultimate substance, if it is not in contact with any.

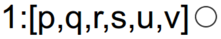
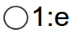
LHS	RHS	NAC
		

Similarly there are 10 more layers of rules in this generative grammar. The layers from 8 to 16 are briefly described here, but presented in detail in [9].

The 8th layer is to generate qualities in mobile and ubiquitous ultimate substances whereas the 9th layer is to generate qualities in wholes. The next few layers (10th to 15th) have rules devoted to the generation of nodes standing for universals. The 16th layer is to generate UDs in USs – both mobile as well as ubiquitous.

The 17th and the final layer is the one where all the label names (except that of contact and disjunct) will be replaced with a common label *e*. This is a way of erasing all the label-based identities of various nodes, and also creating an occasion to prove later, using the interpretative rules, that the different categories of nodes in the graph can be identified purely based on their structures/forms and not their labels.

This layer has only one rule and this is applied only on the TGs of 16th layer.

LHS	RHS
	

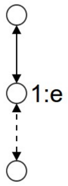
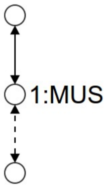
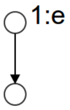
3.2. Interpretative Rules of Vaiśeṣika

The difference between generative rules and interpretative rules of Vaiśeṣika can be thought of as the difference between syntax and semantics of reality. The graphs generated by the former – generative rules - refer to the syntactic portion of reality – they are forms or arrangements of entities in reality. Given the way entities are arranged we discern their meaning and categorize or classify them accordingly, and that, the semantic portion of reality, is given by the latter – interpretative rules.

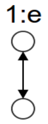
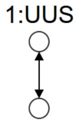
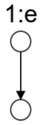
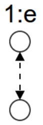
Interpretative rules are also graph grammars like generative rules, but we separated both of them because their purposes are different – one parses the graphs which the other generates. So the interpretative rules can be thought of as constituting an interpreter or a parser for the language (graphs) of Vaiśeṣika which is generated by the generative rules in the previous subsection.

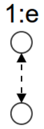
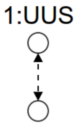
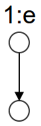
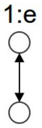
So the input to the interpreter are the TGs generated by the last layer (17th layer) of the generative grammar. Any of them can become the start graph for the interpretative rules (IRs). Coming to the alphabet, the edge-labels of interpretative rules (Ω'_E) are same as that of generative rules (Ω_E) i.e. $\Omega'_E = \Omega_E$ whereas the node-labels of interpretative rules include the node-labels of the TGs of GRs, as well as Vaiśeṣika categories i.e. $\Omega'_V = \{e, C, D, U, UD, SW, Q, MUS, UUS\}$. In some rules nodes are unlabeled. Such anonymous nodes stand for any node with any label. Also the IRs are not prioritized the way GRs were. IRs can be applied in any order since they are independent of each other. The process completes when there is no scope left for any rule to be applied. These IRs can also be considered as a graphical way of defining the categories of Vaiśeṣika. The rules are given below.

The first rule defines a Mobile Ultimate Substance (MUS). An MUS is one which inheres nowhere (NAC) and is in contact with at least one entity and is in disjunct with at least one entity (Definition 2.15).

LHS	RHS	NAC
		

The 2nd and 3rd rules define a Ubiquitous Ultimate Substance (UUS). A UUS is one which inheres nowhere (NAC$_1$) and is only in contact but not disjunct (NAC$_2$) with any other entity (2nd rule) (or) only in disjunct but not contact (NAC$_2$) with any other entity (3rd rule) (Definition 2.14). They are shown in the next two tables respectively.

LHS	RHS	NAC$_1$	NAC$_2$
			

LHS	RHS	NAC$_1$	NAC$_2$
			

Similarly we have interpretative rules to define the remaining categories like UD, U, SW and Q. They are presented in [9], the extended version of this paper.

4. Results

We have constructed a graph grammar software of our own to simulate the above generative as well as interpretative rules, and on around 10000 terminal graphs of the last layer of generative rules, we ran the interpretative rules and found that we could interpret every node of every graph there as one of the Vaiśeṣika categories. This gives us a confidence that the system is not only intuitively sound but also inductively sound, and that the structure of reality, as described in Vaiśeṣika, can be discovered, generated and parsed purely formally. This shows that the categories are differentiated purely based on their formal structures and nothing else. The conclusive proof of soundness of the system is part of our future work.

5. Conclusion

The idea is to present an ontology language which is graph-based and which can be formal and which shows the potential to be scaled up in future to cover many other portions of reality like causation, cognition, language etc. We think we have accomplished that job in this paper by taking Vaiśeṣika ontology as an example, building generative as well as interpretative rules to derive the valid sentences (graphs) of Vaiśeṣika and showed that a rigorous formal ontology is possible. Since the focus of this paper is on generative and interpretative grammars of Vaiśeṣika and not on the rationale behind its categorial system there is no scope for comparing this with other existing ontologies since, in our knowledge, there is no other foundational ontology which has grammar(s) like ours. The only comparison we can draw with other ontologies is that they are (semi-)manually constructed while ours is generated. Here only the grammar is manually written but the actual ontological graphs are computationally generated. In other words potentially infinite structures can be generated using a finite set of rules in our ontology whereas in others they need to actually come up with the potentially infinite structures (semi-)manually.

6. Future Work

As mentioned earlier this paper is only the first step toward building a formal ontology which envisages to cover a major portion of reality – a long-term project in its own right. We have covered only the locational contiguum, that too some portion of it, in this paper, and one could see how rigorous the study of even such a small portion can be. This is purely a theoretical work and for any fruitful applications to come out of this kind of work, one needs to formalize the remaining contigua, at least till the 4th one i.e. expression contiguum. Our immediate priority is to prove the soundness of the system we presented till now in a foolproof manner, and then continue to extend the system till we formalize expression contiguum.

References

[1] Kanada. *The Vaiśeṣika sutras of Kanada*, with the commentary of Samkara Misra and extracts from the gloss of Jayanarayana. Translated in English by Nandalal Sinha. Allahabad (1911); 2nd edn. Revised and enlarged, Allahabad (1923); Reprinted New York (1974), Delhi (1986)

[2] Prasastapada, *Padarthadharmasamgraha* with Nyayakandali of Sridhara. English Translation by Ganganatha Jha. Chowkhamba, Varanasi (Reprint 1982)

[3] Udayana, *Lakshanavali*. In Musashi Tachikawa, The Structure of the World in Udayana's Realism: A study of the Lakshanavali and Kiranavali. Springer, Heidelberg (1982)

[4] Singh, N. Comprehensive Schema of Entities: Vaiśeṣika Category System. 2001. *Science Philosophy Interface* 5(2): 1–54.

[5] Singh, N. *Formal Theory of Categories through the Logic of Punctuator* (2002) (unpublished)

[6] Singh, N., Theory of Experiential Contiguum. 2003. *Philosophy and Science: Exploratory Approach to Consciousness* 111–159, Ramakrishna Mission Institute of Culture, Kolkata.

[7] Singh, N. *Foundations of Ontological Engineering*, Lecture slides of course at IIIT Hyderabad (2008)

[8] Rozenberg, G., et al. eds. 1997. *Foundations. Handbook of Graph Grammars and Computing by Graph Transformation, vol. 1*. Singapore: World Scientific.

[9] Tavva, R., and Singh, N. *Generative Ontology of Vaiśeṣika*. https://sites.google.com/site/vrktavva/resources/Generative_Ontology_of_Vaiśeṣika.extended_version.pdf?

[10] Leibniz, G.W. *The Labyrinth of the Continuum: Writings on the Continuum Problem, 1672-1686*. Translated from Latin and French to English by Richard T.W. Arthur. New Haven: Yale University Press (2001).

[11] Brentano, F. *The Theory of Categories*, translated from German by Roderick M. Chisholm and Norbert Guterman. The Hague, Boston, London: Martinus Nijhoff (1981).

[12] Roberto, P., et al. eds. 2010. *Theory and Applications of Ontology: Computer Applications*. Springer.

[13] Comprehensive list of Vaiśeṣika Categories. https://sites.google.com/site/vrktavva/resources/Vaiśeṣika_Categories_table.pdf

[14] Mukhopadhyay, P.K. 1984. *Indian Realism: A Rigorous Descriptive Metaphysics*. Calcutta: K P Bagchi & Company.

[15] Casati, R., & Varzi, A. C. 1999. *Parts and places: The structures of spatial representation*. Cambridge, Mass: MIT Press.