Formalizing Narrative Structures: Glossematics, Generativity, and Transformational Rules

Francesco Galofaro
1. Purpose of the article

In this essay, we will test the material adequacy of the models proposed by early structuralist and formalist scholars, such as Hjelmselv and Jakobson, regarding the relation between system and process. In particular, Hjelmslev’s glossematic theory is too weak: it mechanically generates just every kind of symbol combinations, but we need a grammar capable of generating both narrative enunciates and their structural description. On the other hand, a model such as Propp’s folktales morphology is too restrictive, because it does not take into account a fundamental aspect of narrative structures: recursiveness.

In order to demonstrate this point we will refer to Chomsky’s classification of formal grammars from Turing Machines to Finite-state automata, each generating its specific kind of language. We will also use Chomsky’s formal techniques to sketch a generative grammar for narrative structures. Our purpose is limited to frame recursiveness, a fundamental property in the link between system and process which is not represented in semiotic metalanguage. Every recursive definition of a structural element represents an actual procedure (in the Hjelmslevian sense) describing the articulation of meaning. In conclusion, conscious of the limits of the techniques employed to discover the features of the deep narrative structure, such as recursiveness, we will argue in favour of morphogenetic research on recursiveness as a property that assured structural stability to narrations of growing complexity in a diachronic perspective.

2. Hjelmslev: between classification and generation

Hjelmslev’s work marks the transition between two different conceptions of grammar which were popular at the time. According to the traditional concept, the task of describing language may be considered comparable to the segmentation of the linguistic stream and the classification of its corresponding units. Hjelmslev borrowed his second point of view regarding language from the logicist tradition — cf. Carnap (1934) — according to which language may be described as a calculus, with its own set of rules regarding the formation and transformation of a particular sentence. For example, a *transformation rule* can describe the relation between the active and the passive form of a sentence: *if* Mary loves John *then* John is loved by Mary. The theory consists of analytical relationships as well: in glossematic terms, *if* $x$ is a purport, *then* $x$ is a class of variables — cf. Hjelmslev (1973).

Apparently Hjelmslev had an open mind with regards to the notion of calculus. Glossematics is partially formalized and provides a set of symbols, some specifics regarding the relationships between them, and some rules for analysis. The link between linguistics and calculus is the commutation test. In other words, Hjelmslev applied Ajdukiewicz’s *substitution* (1935) to linguistic elements smaller than a “sign”. In Ajdukiewicz’s work, this technique allows logicians to describe the categorial grammar of natural languages. It also allows us to perform a calculus, thanks to which we can decide whether a sentence is grammatical or not (a decision problem). This method was used for the first time in Husserl’s fourth logical inquiry (1922).

Hjelmslev borrowed his logistic point of view from research programs as the finitist studies conducted by Hilbert and Carnap: he believed in a finite foundation of knowledge on a general schema, his glossematic theory:

In deference to the simplicity principle, metasemioologies of higher orders, on the other hand, must not be set up, since, if they are tentatively carried out, they will not bring any other results than those already achieved in the metasemiology of the first order or before — Hjelmslev (1943, eng. trans. p. 125).

In Galofaro (2012a) I put forward a formal argument, similar to Turing’s *Entscheidungsproblem*, in order to demonstrate that Glossematics, as described by Hjelmslev (1975), is an “essentially incomplete” theory. “Essentially incomplete” meaning that, if Glossematics is a coherent theory, then it is necessarily incomplete. In particular, if Glossematics is a coherent theory, it is incapable of demonstrating

---

2. The technique which segments the strings of every semiotic process by replacing the segment of the expression plan with another, and by observing changes on the content plan (the meaning). The segmentation process continues until we register units which have no relationships with the content plan. In this manner we are also able to classify the semiotic elements and their relationships (figures, constants, variables, cenematemes, pleremes, and so on). Thus Glossematics considers the process chain (the realized text) as a particular combination of elements.
its completeness. According to the essay, Hjelmslev was wrong: Semiotics cannot provide the basis of knowledge utilizing finitists methods. His theory cannot withstand the arsenal other logicians developed to thwart attempts to, once and for all, base mathematics and scientific knowledge on a logical metalanguage. He was, after all, a linguist and not a logician. Nevertheless, even if it is false that Semiotics does not need any other finite foundation besides itself, this does not mean that our discipline is incapable of describing other types of semiotic structures. The logical empiricist dream may have vanished; however, many of its key concepts and techniques may be used in a laical perspective. For example, generative linguistics is principally based on calculus; on the same line, in its effort of formalization, semiotics obtained some results, which can be used in the field of narrative structures. In particular, we will ascertain whether Hjemslev’s framework is adequate to the task of generating narrative structures, and we will consider the alternatives.

As we’ll see, another necessary and interesting feature of Hjelsmlev’s epistemology is the metalinguistic approach. It is simpler to generate the symbolic, inter-defined, univocal metalanguage Greimas created to describe a text, than to produce the entire text in its every detail, even those details which are not relevant to the narrative structure.

3. Automatic grammars

In Galofaro (2012) I used a mathematical technique known as gödelization to implement calculus in semiotic metalanguage. Thanks to it, one may encode glossematics by associating each symbol, including relations, to a unique number. In this manner, a formula becomes a string of Gödel numbers and vice versa. The same technique may be used to encode the entire Glossematics theory, considering each formula-number as a stand-alone symbol. Once Glossematics has been completely encoded, operations included, one may transform some formulas into others via mathematical operations, respecting the logical syntax of

---

3. According to Hjelsmlev (1943), the whole semiotic theory is an analysis. Thus one could ask what happens when the theory analyses itself: Hjelmslev admits this possibility and claims that semiotic theory does not need any other foundation than itself. In Galofaro (2012a) I show how a semiotic analysis can be encoded as a software. Then I imagine a software m which analyses a semiotic analysis a of an input x (ma ) and returns the values: [0 = a reaches a conclusion; 1 = a does not reach a conclusion]. Now, let’s say that the input x is a, and let’s imagine the software T which reaches a conclusion and alerts us if ma does not reach a conclusion. Unfortunately, T does not reach a conclusion if ma reaches a conclusion. Now, what happens to M? What happens when the theory analyses itself? This software should reach a conclusion when it does not reach a conclusion and vice-versa. Hence, in principle such a program cannot be written. This feature of semiotic theory is useful to demonstrate the presence of undecidable formulas.

4. It is the technique developed by Gödel in order to demonstrate his two famous theorems. Cf. Gödel (1986).
the theory. For example, the property “being an analysis of” may be represented as a relationship \( R \) between the Gödel number of the whole theory and the Gödel numbers representing its elements. Consequently, any coherent semiotic formal model can be represented as a Turing Machine.

In brief, what is a Turing Machine? It is the simplest type of computer. It is composed of an infinite tape separated into squares, each of which can contain a printed symbol (its memory). The machine scans a symbol, matches it to an instruction table, and performs an action. It can erase the symbol, however this is not necessary; it can replace it with another; it can move the scanning device one space to the left or right; it can assume the same state or a new one. Hence, a universal TM could generate any type of string by combining a given alphabet. Exactly as the famous ape playing with a typewriter, it could generate any sort of text, from Tolstoy’s novels to completely meaningless combinations, not to mention a Dan Brown fiction novel.

How does Hjelmslev’s system work? It works just fine, thanks. All jokes aside, according to Hjelmslev, combinations in the syntagmatic axis are marked by the function “and”, whereas the relationship between the system’s elements is marked by the relation “or”. Hence, the system includes all the categories \{A, B, C \ldots N\} where each category is a finite set of elements — for example, if A is a category, then \( A: \{a_1, \text{ or } a_2, \text{ or } a_3, \ldots \text{ or } a_n \} \). On the other hand, the generated processes are finite chains of elements selected from each category, so \( P: \{ \text{a}_x \text{ and b}_x \text{ and c}_x \ldots \text{ and n}_x \} \).

Unfortunately, structuralist schemas may be considered somewhat naive regarding the relation between system and process. For example, in Jakobson’s terms, the system is the domain of the selection function, whereas a simple combination function generates the process; even in the discussion between Propp and Levi-Strauss — in Propp (1984) — the ethnologist depicts the narrative system as a combination of elements, wherein, sooner or later, every concatenation will be attempted.

All these combinations may be performed by a simple Turing Machine. But is this a Grammar model? As Chomsky (1959) states, by generating every sentence in a language and placing it in a numbered file, Turing Machines say nothing of interest about the grammatical features of a given sentence: the results are trivial. On the contrary, a grammar should assign a structural description to every generated string indicating how relevant structural information can be obtained for every string generated by the theory. In other words, early structuralist models are too weak.

### 4. A Portrait of Glossematics as a finite-state automaton

In order to describe the structure of the generated concatenation, we will adopt a “machine” which mechanism can assume a finite number of internal states. These states are connected in pairs, in order to express the rules of syntax: not every
combination of categories is allowed, nor does every chain between elements derive from them. The automaton is linked to a printer device and, for each transition between connected states, it types a symbol which derives from the elements of each category. This type of automaton can produce syntagmatic chains, such as the one described by Hjelmslev. As we deal with a finite-state automaton, we will call the system which generates these structures a finite-state system.

Now, in order to test the adequacy of this automaton, we will determine whether it can generate every sort of narrative structures. The simplest example of a finite-state system generating narrative structures is Propp’s Morphology of the folktale (1928a). In Propp’s model, there are 31 syntagmatic functions which form chains ordered in a precise succession. For example, the interdiction and the violation of the interdiction always precede the departure function; the villainy or the lack always precede the acquisition of the magical agent, and so on. This syntagmatic succession may be generated by a finite-state system. As we mentioned previously, the system has a finite number of internal states. The links between narrative functions are represented by connections between pairs of states. Each time the system passes through two connected states it produces a symbol, which corresponds to a function in Propp’s model. In order to produce a narrative structure, this system begins with the initial state and passes through the series of connected states producing one of the symbols associated to each state, thus generating every possible narrative structure of the fairy tale. Fairy tales have a finite-state narrative structure. In fig. 1 we can see an example of an automaton which produces a simple folktale plot.

![Fig. 1. A finite-state automaton generating a simple plot](image)

Chomsky (1956) exemplifies some non-finite-state structures, which cannot be generated by a finite-state automaton. For example, we can picture a set containing the processes aa, bb, abba, baab, aabbaa, baaaab and so on, generated by a rule which adds to a string x its specular reflection. Do we find such a narrative structure in our everyday practice? The answer is yes. I recall the distinction

---

5. Chomsky (1956) demonstrates that such an automaton can’t even generate all the possible language structures.
Greimas drew between base-narrative programs and use-narrative programs. In order to accomplish a certain task, the protagonist has to execute a number of base-narrative programs. Every base-narrative program can consist of a sequence of use-narrative programs: so, if the base-narrative program is “to cook a piece of meat”, I have “to turn on the fire”, “to put some oil and rosemary in the saucepan”, “to put the meat in the boiling oil”, “to add a glass of white wine”, “to put the cover on the pot”, “to wait for 15 minutes”, “to add some potatoes”, “to wait for 45 minutes” and so on. Each use-narrative-program can be further subdivided into new use-narrative-programs; for example, “to add a glass of white wine” implies “to search for wine bottles in the cellar”, “to select a white wine bottle”, “to remove the cork”, “to verify if the wine is still drinkable” and so on. Traces of a recursive function may also be found in Propp’s model, where he states that a function can be triplicated in the narrative process: for example, in order to acquire the magical agent, the hero has to pass three tests. These narrative structures may be used in order to produce specular structures such as:

\[
\text{Base NP}_1 \quad \text{Base NP}_2
\]

\[
\text{Use NP}_{1.1} \quad \text{Use NP}_{1.n}
\]

Fig. 2. Hierarchy of narrative programs

These structures form processes similar to the ones described by Chomsky. In other words, narrative structures, just as grammatical ones, are recursive and cannot be generated by a finite-state system. We need to adopt a more complex model of the relation between system and process if we want it to be sufficiently adequate to generate these narrative structures. All things considered, it is somehow apparent that Propp’s system generates only fairy tales, but not, say, sci-fi novels, as did the electrical bard of the famous short story “Someday”, by Isaac Asimov.

---

6. Lakoff (1972) attempts to demonstrate that even Propp’s model is not a finite-state automaton. In my opinion, this is not correct. Propp’s model does not describe every type of fairy tale, but only the typical Russian folk tale about magic — cf. Propp (1984). Not every type of fairy tale can be generated by a finite-state automaton, but Propp’s does. The empirical evidence is the large number of fairy tale generators that can be found on the net (e.g, http://www.stonedragonpress.com/vladimir_propp/propp_generator_v1.htm).
5. Aims of a generative-transformational model

We demonstrated how Hjelmslev’s glossematics is inadequate in describing narrative structures because it does not consider recursiveness. This is not surprising: when Hjelmslev wrote his books, the study of these functions was only in its early stages; nevertheless Hjelmslev had an interesting intuition, which anticipated Chomsky’s works: instead of pretending that the structure of a language is similar to formal logic, the latter could aid in the construction of an adequate metalanguage.

However, here is a question which begs an answer: why should we formalize semiotics? Formalization is an antiquated dream — or nightmare — of sorts. Beginning with Greimas, many scholars were hopeful regarding the possibility of formalizing semiotics. Nevertheless, the simple idea of automatizing semiotic analysis is absurd: it is a parody of the project’s actual goals. Furthermore, I doubt that the entire generative path, as Greimas described it, may be formalized utilizing just one technique. Lastly, formalization does not prove anything regarding the “accuracy” of our models, nor about their “reality”, because there are many different possible, coherent and adequate formal descriptions of every phenomenon — cf. Putnam (1988, pp. 95-96, 121-125).

Where does this leave us? We simply need a clear, effective procedure which assigns, without ambiguity, each element of our structure to its category, its place within the system, specifying the link between system and process. If this cannot be done in principle, then terms such as system, process, structure and enunciation, should be completely banned, because it is impossible to clarify how to use them in an unambiguous manner, thus transforming semiotics into scientism, or into a pedantic form of hermeneutics. In conclusion, we will also discuss the limitations of the techniques we have been using to sketch our model.

6. A generative model of narrative syntax

In order to describe a system which generates a formal metalanguage representing narrative structures, we first need a metalanguage. We will use the one elaborated by Greimas and by his school, for two reasons: one, because it is generally accepted, with a few exceptions, in the semiotic community, whereas other theories did not gain popularity; and because it is formal. Here is a brief summary.

According to Greimas, in a narrative situation, subject S is or is not conjoint to a valuable object (Ov). Simondon (2005) would most likely identify these constructions with “metastable situations”: (S∩Ov) or (SUOv). A transition between two metastable states is always possible: a destinant (D) expresses the

---

condition which lets the subject become conjoint or disjoint with a valuable object:

EN: \( D \rightarrow (S \cap Ov) \) or
EN: \( D \rightarrow (SUOv) \)

Typically a complete narrative program (NP) is a chain between enunciates describing the transition between metastable states. In this metalanguage, terms as “subject”, “object”, “destinant” are not referred to persons or things: they are labels for narrative functions. For example, the valuable object could be a princess from the point of view of the paladin; a treasure, in the case of a pirate; the solution of an equation, if the protagonist is a mathematician; “socialism” if the subject is interpreted as a revolutionary leader.

In order to provide an example, I will summarize Marsciani & Zinna’s analysis of Genesis. God puts Adam and Eve in the Garden of Eden and — in exchange — he tells them that eating an apple is forbidden:

\[
\begin{align*}
\text{EN}_1: & \quad [D_1 (\text{God}) \rightarrow (S_1 (\text{Adam and Eve}) \cap Ov_1 (\text{Eden}))] \\
\text{EN}_2: & \quad [D_2 (\text{God}) \rightarrow (S_2 (\text{Adam and Eve}) \cup Ov_2 (\text{apple}))]
\end{align*}
\]

We can notice how the role of the subject is played by a collective actor, the role of the destinant by a transcendent entity, and the object of value by a space. Now, Adam and Eve violate the contract (in a complex way which implies Use-NP that we will not consider here).

\[
\begin{align*}
\text{EN}_3: & \quad [D_3 (\text{serpent}) \rightarrow (S_3 (\text{Adam and Eve}) \cap Ov_3 (\text{apple}))] \\
\text{EN}_4: & \quad [D_4 (\text{God}) \rightarrow (S_4 (\text{Adam and Eve}) \cup Ov_4 (\text{Eden}))]
\end{align*}
\]

So God punishes them:

\[
\begin{align*}
\text{EN}_5: & \quad [D_5 (\text{God}) \rightarrow (S_5 (\text{Adam and Eve}) \cap Ov_5 (\text{Eden}))] \\
\text{EN}_6: & \quad [D_6 (\text{God}) \rightarrow (S_6 (\text{Adam and Eve}) \cup Ov_6 (\text{Eden}))]
\end{align*}
\]

Unlike Greimas, we gave each actant an appropriate index, in order to assign each of them to their enunciate. After all, as Greimas and Courtés (1979, “actor”) state, we can recognize the identity relation between the subjects of two distinct narrative enunciates if and only if there is a function which assigns them to the same actor. The actor anaphorically crosses the limits of the single enunciate and ensures a semantic coherent layer to the discourse. For example, in the story of Adam and Eve, we can say that \( Ov_4 = Ov_1 \) only because they are both actionalized by /Eden/, elsewhere nothing would assure their identity.\(^8\) Naturally, the same actors (Adam and Eve) may be assigned to more than one actantial function during the narrative structure. We will not consider the conversion between actants and actors as a part of the generative grammar. As we will see, the complete deep structure generated by the generative component of the grammar will be the input of the transformational component whose output is the surface structure of the discursive level. Furthermore, our sketch does not consider other semantic relationships between the various actors (e.g. the apple is in the garden of Eden) In

\[\text{8. As Greimas noticed, the link between actants and actors poses a problem of individuation — cf. “actor” in Greimas & Courtés (1979).}\]
other words, this grammar does not explain the entire generative path described by Greimas and Courtés: it is simply a generative model for the narrative syntax, a metalanguage of empty positions for supplementary semantic investments.

6.1. A formal, generative system for narrative structures

All things considered, we wish to describe a generative system that generates every enunciate of a particular narrative program in the correct order. First of all, we will say that our syntagmatic system contains a symbolic repertoire and some derivational rules. The derivational rules have the form \( (\alpha \vdash \psi \lor \omega) \), where \( \alpha, \psi \) and \( \omega \) are arbitrary symbols and \( \vdash \) represents a substitution. Each rule substitutes just one symbol with a second one or with a couple of symbols. The symbolic repertoire consists of

a) the initial symbol NP;

b) the intermediate symbols \{be, let, \( \mathfrak{N} \), EN, J\}

c) an index \( x \) (a counter).

d) the terminal symbols \{S, D, U, Ov, \( \cap \), \( \rightarrow \)\}

Terminal symbols are the ones used by Greimas in his attempt to formalize narrative structures. Let’s clarify the sequent rules, thanks to which we can outline the derivation of a string of terminal symbols from the initial symbol.

\[
\begin{align*}
1) & \text{NP}_x \vdash \mathfrak{N}x \\
2) & \text{N}x \vdash \text{EN}_x, \text{possibly N}_{x+1} \\
3) & \text{EN}_x \vdash \text{let}_x, \text{be}_x \\
4) & \text{let}_x \vdash \text{D}_x, \rightarrow_x \\
5) & \text{be}_x \vdash \text{S}_x, \text{J}_x \\
6) & \text{J}_x \vdash \cap_x \text{or U}_x, \text{Ov}_x
\end{align*}
\]

This grammar is not complete. We will add some conversion rules which describe how each narrative functive and function is interpreted by discursive elements, narrative structures and discursive structures for each function (as \( D_x \vdash \text{“Serpent”} \)). Let us focus for a moment on rule 2: it ensures grammar recursivity. There is an aleph-element (\( \mathfrak{N} \)) which generates every enunciate we need through the incrementation of its counter. It represents the property of “being a recursive element”, which is owned by every element after it. Rule 2 also ensures that our system is self-embedding.\(^9\) This is another interesting feature of narrative structures: according to Chomsky (1959) a language cannot be considered as a finite-state language if, and only if, all the systems which generate it are self-embedding. Recursiveness is a general property of narrative structures: if a text is composed of an NP chain, we can generate the exact chain with the same technique, using the \( \mathfrak{N} \)-elements.

\(^9\) A system is self-embedding if, for a symbol \( \alpha \), it contains a rule as \( \alpha \vdash \psi \alpha \omega \).
6.2. An example: the Genesis

In order to understand how the system works, let us provide here an example of the derivation of a single narrative enunciate, representing the serpent which gives Adam and Eve an apple:

\[ \text{N}_1 \text{(rule 1));} \]
\[ \text{EN}_1 \text{(rule 2));} \]
\[ \text{let}_1 \text{ be}_1 \text{(rule 3));} \]
\[ \text{D}_1 \rightarrow_1 \text{ be}_1 \text{(rule 4));} \]
\[ \text{D}_1 \rightarrow_1 \text{ S}_1 \text{J}_1 \text{(rule 5));} \]
\[ \text{D}_1 \rightarrow_1 \text{ S}_1 \text{J}_1 \text{Ov}_1 \text{(rule 6));} \]

In the last row we only have terminal symbols, so the derivation ends. Thanks to it, we have generated a narrative enunciate. According to Greimas’s interpretation of the terminal symbols, a destinant D lets a subject S be conjoint with a value-object O, thus describing the narrative structure of Genesis 3,1-6. As we mentioned previously, the passage in Genesis is composed of use-narrative programs, and should be described recursively. We will discuss how this may be done later on.

Let’s add some ad hoc rules which will ensure an interpretation of the general narrative structures we derived:

a) \[ D_1 \rightarrow \text{God} \]
b) \[ \rightarrow_1 \text{ allows that} \]
c) \[ S_1 \rightarrow \text{Adam and Eve} \]
d) \[ \cap_1 \text{ join} \]
e) \[ Ov_1 \rightarrow \text{Eden} \]

Now we can continue the derivation in this manner:

\[ \text{God} \rightarrow_1 \text{ S}_1 \text{J}_1 \text{Ov}_1 \text{(rule a));} \]
\[ \text{God allows S}_1 \text{J}_1 \text{Ov}_1 \text{(rule b));} \]
\[ \text{God allows that Adam and Eve S}_1 \text{J}_1 \text{Ov}_1 \text{(rule c));} \]
\[ \text{God allows that Adam and Eve join Ov}_1 \text{(rule d));} \]
\[ \text{God allows that Adam and Eve join Eden (rule e));} \]

In the same way we could obtain “God interdicts that Adam and Eve eat the apple”. We must avoid any confusion of these descriptions with the textual surface, or with its discursive structure: they are simply labels that interpret the generated structure.

Let us draw the derivation diagram:
Now we will comment on the diagram. It becomes clear that Greimas’s parenthesis marked deep structural relationships: the linear aspect of the surface is just an illusion. Modal functions such as “let” and “be” govern the surface of the enunciate. We can also affirm that the label “join” interprets the conjunction \( \cap \), which is a kind of junction \( J \), governed by a “be” modal function which is part of the first narrative enunciate \( EN \). By using only binary branches, we present the system in a regular form utilizing Chomsky’s terminology (1959); in this manner all the constituents of the enunciate are attributed in a univocal way and the structure is described in its entirety, even if the \( N \)-elements generate a virtually infinite thread. This is one way to formulate recursive definitions in semiotics: an element or a concept is not defined through its link with a category and its specific difference, but through the indication of an effective procedure through which it is generated.

Let us briefly discuss indices. They will aid us in our discussion of recursiveness, because through them, we can track the hierarchical relationships between the syntagmas of the terminal string. I plan on using them to formalize the transformational component of grammar in a future essay. In Greimas (1976) they are used in a different way: they mark the identity of a functive through the narrative structure. For example, if the first EN assigns to “Eve” the symbol S2, this number will distinguish her from other subjects even in the other ENs. In our model, narrative identities are determined as the result of the application of...
“attribution” rules as (a … e), which take place during the conversion between the narrative structure and the discursive one. It would be relatively simple to modify our grammar to represent the trans-enunciative identity of the actants. Nevertheless, this choice would be questionable under many aspects. I have already presented some of them in paragraph 5: what helps us distinguish and individuate the persistence of a function through the narrative structure is the fact that it is assigned to a particular “actor”, in other terms to a discursive element which constitutes a coherent layer — in the words of Greimas: an isotopy. Furthermore, surface-units often change their narrative function.

Now we will examine a more complex derivation which generates Marsciani & Zinna’s analysis of Genesis. In order to avoid a long and boring process, we will summarize rules 5-7 this way:

5a) \( \text{be}_x \xrightarrow{} (S_x \cap S_{Ov_y}) \text{ or } (S_x \cup S_{Ov_y}) \); 

Likewise, in the corresponding diagram we will use a triangle in order to represent that our final structures are not the simplest possible structures (a common convention in linguistics).

1) \( \text{EN}_1, \text{EN}_2 \) (rule 1);
2) \( \text{EN}_1, \text{EN}_2 \) (rule 2);
3) \( \text{EN}_1, \text{EN}_2, \text{EN}_3, \text{EN}_4 \) (rule 2);
4) \( \text{EN}_1, \text{EN}_2, \text{EN}_3, \text{EN}_4 \) (rule 2);
5) \( \text{EN}_1, \text{EN}_2, \text{EN}_3, \text{EN}_4 \) (rule 2);
6) \( \text{let}_1 \text{be}_1, \text{EN}_2, \text{EN}_3, \text{EN}_4 \) (rule 3);
7) \( \text{D}_1 \xrightarrow{} \text{be}_1, \text{EN}_2, \text{EN}_3, \text{EN}_4 \) (rule 4);
8) \( \text{D}_1 \xrightarrow{} S_1 \cap S_{Ov_y}, \text{EN}_2, \text{EN}_3, \text{EN}_4 \) (rule 5a);
9) \( \text{D}_1 \xrightarrow{} S_1 \cap S_{Ov_y}, \text{let}_2 \text{be}_2, \text{EN}_3, \text{EN}_4 \) (rule 3);
10) \( \text{D}_1 \xrightarrow{} S_1 \cap S_{Ov_y}, \text{D}_2 \xrightarrow{} \text{be}_2, \text{EN}_3, \text{EN}_4 \) (rule 4);
11) \( \text{D}_1 \xrightarrow{} S_1 \cap S_{Ov_y}, \text{D}_2 \xrightarrow{} S_2 \cup S_{Ov_y}, \text{EN}_3, \text{EN}_4 \) (rule 5a);
12) \( \text{D}_1 \xrightarrow{} S_1 \cap S_{Ov_y}, \text{D}_3 \xrightarrow{} S_2 \cup S_{Ov_y}, \text{let}_3 \text{be}_3, \text{EN}_4 \) (rule 3);
13) \( \text{D}_1 \xrightarrow{} S_1 \cap S_{Ov_y}, \text{D}_2 \xrightarrow{} S_2 \cup S_{Ov_y}, \text{D}_3 \xrightarrow{} \text{be}_3, \text{EN}_4 \) (rule 4);
14) \( \text{D}_1 \xrightarrow{} S_1 \cap S_{Ov_y}, \text{D}_2 \xrightarrow{} S_2 \cup S_{Ov_y}, \text{D}_3 \xrightarrow{} S_3 \cap S_{Ov_y}, \text{EN}_4 \) (rule 5a);
15) \( \text{D}_1 \xrightarrow{} S_1 \cap S_{Ov_y}, \text{D}_2 \xrightarrow{} S_2 \cup S_{Ov_y}, \text{D}_3 \xrightarrow{} S_3 \cap S_{Ov_y}, \text{let}_4 \text{be}_4 \) (rule 3);
16) \( \text{D}_1 \xrightarrow{} S_1 \cap S_{Ov_y}, \text{D}_2 \xrightarrow{} S_2 \cup S_{Ov_y}, \text{D}_3 \xrightarrow{} S_3 \cap S_{Ov_y}, \text{D}_4 \xrightarrow{} \text{be}_4 \) (rule 4);
17) \( \text{D}_1 \xrightarrow{} S_1 \cap S_{Ov_y}, \text{D}_2 \xrightarrow{} S_2 \cup S_{Ov_y}, \text{D}_3 \xrightarrow{} S_3 \cap S_{Ov_y}, \text{D}_4 \xrightarrow{} S_4 \cup S_{Ov_y} \) (rule 5a);

Again, in the last row we find only terminal symbols, so the derivation has come to an end. We can see that the application of rule number 2 recursively generates all the enunciates we need. The limit, if it exists, is posed by the text we want to analyze, not by the system.

10. Greimas represents a similar function through the use of colon, and the result is that his indexes and elements marks pleonastically the identity of narrative functives.
Perhaps my reader could ask if aleph generates temporal order. This is indeed a good question. The deep structure presents the events and the actions in a logical order. This order is a hierarchy (see fig. 4). According to Greimas’s theory, temporalization is a feature of the surface-structure. I agree with this choice: the logical structure does not necessarily coincide with the chronological order, as in science-fiction novels, in fantasy books, and in Einstein’s theory of relativity.

In order to understand the immanent structural dependencies between the generated elements, we can construct the following diagram:

![Fig. 4. Genesis as a recursive narrative structure](image)

The indexes 1, 2, 3 and 4 mark the different phases of the narrative program. In different situations, they could represent the canonical narrative path in Greimas’s terms: an initial contract between S and D; S acquires a competence; S executes a performance; depending on the results of the performance, S is positively or negatively sanctioned by D. Another interesting property of these structures is teleology. Aleph is not a terminal symbol, so it needs to be substituted by another

---

11. In the structural and post-structural tradition, “immanence” is opposed to “manifestation”. Manifested textual meaning is generated through immanent structures. This way it avoids a direct epistemological foundation on this or that particular psychological or philosophical theory, which can rapidly become old-fashioned. Cf. “Immanence” in Greimas & Courtés (1979).
string. This means that the structures with a higher index presuppose the ones with a lower one: cf. “Elements pour une grammaire narrative”, 3.3.3, in Greimas (1970).

6.3. Recursive structures: how to cross the Jordan

Now you may wonder, what about the recursive functions which generate use-programs starting from base-programs? As we stated previously, it is very difficult to represent Genesis 3, 1-6 with a single enunciate. It can be more adequately represented by a use-narrative program. This can be done by adding a single rule to our grammar:

\[ EN_y \vdash NP_{y,1} \]

In this manner we can represent Use-narrative program specifying, thanks to the index, that they are governed by a particular EN, and rules 1-5 can be applied to them recursively. We will obtain diagrams with branches such as this:

Fig. 5. A representation of the recursive relation between a base-narrative program and a use-narrative program

In order to convince my reader, I will apply the grammar to a different example: the crossing of the Jordan (Joshua 3-5). Joshua must take the children of Israel across the Jordan in order to succeed Moses as leader (Base PN). Step by step, God transfers Joshua the competence to do this:

\[ EN_1; [D_{1,1} (God) \rightarrow (S_{1,1} (Joshua) \cap Ov_{1,1} (competence))] \]

In order to succeed, he has to accomplish a performance, which consists of two different use-PN: to follow the ark across the dry river bed. “And thou shalt command the priests that bear the ark of the covenant, saying, When ye are come to the brink of the water of Jordan, ye shall stand still in Jordan”:

\[ EN_{2.1}; [D_{2.1,1} (Joshua) \rightarrow (S_{2.1,1} (Israel) \cup Ov_{2.1,1} (Ark))] \]
“Take you twelve men out of the people, out of every tribe a man, And command ye them, saying, Take you hence out of the midst of Jordan, out of the place where the priests’ feet stood firm, twelve stones, and ye shall carry them over with you, and leave them in the lodging place, where ye shall lodge this night”.

\[\text{EN}_{2.2}: [D_{2.2} \text{(Joshua)} \rightarrow (S_{2.2} \text{(Israel)}) \cup Ov_{2.2} \text{(erect a monument)}]\]

Finally, God orders Joshua and Israel to perform circumcision, thus removing the reproach of Egypt (positive sanction):

\[\text{EN}_3: [D_{3} \text{(God)} \rightarrow (S_{3} \text{(Israel)}) \cup Ov_{3} \text{(reproach of Egypt)}]\]

I will not draw the derivation, because it is as long as it is tedious. An inquisitive reader may attempt it, and he will discover how easy it can be. Let us look at the corresponding diagram:

How does \( S_1 \) become \( D_2 \)? The answer is in chapter five: as we said, there is not a bijective relation between actors (God, Joshua, Israel … ) and the narrative function (Destinant, Subject, Object). While the narration goes on, the same actor can play different functions.

Any kind of recursive structures can be generated in this way, even the most complex ones. Obviously, by adding new rules the model become increasingly more interesting, in order to adequately describe the text. For example, we could extend the symbolism in order to represent negation (anti-subject; anti-destinant); additional narrative functions as the helper and the opponent; additional modalities. This can’t be done here because our purpose is only to propose a model of generative relations between system and process which can adequately represent recursiveness.
6.4. *The transformational component of the grammar*

By presenting the chaining of narrative enunciates in a hierarchical way, we have discovered a deep-structural element, the $\mathfrak{d}$, which cannot be found at textual surface level but which may explain its recursive organization. If we limit ourselves to the Hjelmslevian conception of the process we would be unable to explain it. This point demonstrates how greatly a formal discussion regarding the type of model we should adopt in Semiotics can still highlight the structural properties which would otherwise remain unnoticed.

Let us now consider two complications. Firstly: our model generates complete narrative structures. Nevertheless, a novel often omits to present one or more phases of the narrative path. One answer could be — as Greimas states — that all the phases are always presumed, even if not always manifested by the surface of the text. How can we formally describe this relation between the system and the process? A second problem relates to the position of the various narrative enunciates at the textual surface level: it is not rare that the text presents them in a different order. The most obvious example is represented by flashbacks in a movie.

In order to solve the aforementioned problems, we have two options. The first is to modify the system, make it more sophisticated, for example adding more symbols and rules. I’m not contrary to this approach in principle. However, for simplicity’s sake, we could admit that the deep narrative structure can be the input of a transformational component of the grammar, which converts it into the surface form, typical of the discursive level. So we should formulate the transformational rules which applies to the former in order to gain the latter.

Putnam (1961) has shed light on an additional problem. One should take care when introducing new rules of transformation, in order to guarantee the recursiveness of the generated structures. Peters and Ritchie (1973) proposed three elementary transformations: the deletion of a sequence, the substitution of a copy of a sequence, and the adjunction of a copy of a sequence to the let and to the right of another sequence. As we have seen, each label of a tree determines a series of brackets which fully describe a terminal element of the sequence. Our indexes should substitute the bracket structure. The transformations are applied cyclically to the strings of the terminal sequence starting from the most internal brackets, to satisfy the *principle of recoverability of deletion*. As Peters and Ritchie wrote, its intuitive meaning is that,

(...) given a speaker who knows the grammar and given a terminal string, the speaker can construct all the structural descriptions for the string generated by the grammar and can furthermore determine that the grammar does not generate the string if this is the case. (p. 69).

An example of a transformational rule could be a “shift” rule. To shift a string of symbols, starting from the proposed elemental transformations, we have to copy it into a different position of the chain and then erase the original one. This way,
a particular narrative syntagma could be shifted after a second one. This “shift” usually leaves a trace on the textual surface, an enunciation mark, as Metz (1991) would call it. For example, the flashback is usually introduced by a dissolve; in certain cases the shifted syntagma is carried out by changes in cinematography. These are popular solutions, but we can obviously expect increasingly complex sophisticated choices in cinematographic language.

We cannot expound upon this argument here, although it is crucial in order to propose a complete model of the conversion between narrative structures and discursive ones: for example, Marsciani (2012) describes how semi-symbolic relationships rule the homologation between levels.

7. Discussion

The main argument of the present essay is that a semiotic theory should represent recursiveness in order to be adequate. For this reason, it is interesting to compare Propp’s model, which is not recursive, and our model. It seems that we are dealing with recursive and non-recursive narrations. How should we interpret this result?

Chomsky believes that language’s structure belongs to the mind. Chomsky’s views regarding the system share some traits with Structuralism: structure does not change over time. Both the cognitive and the structural philosophical frame have been opposed to historicism. Naturally, it is a fact that language changes over time; nevertheless, a look at the languages in use in so-called primitive societies reveals exactly the same structural complexity we find in our culture. Under no circumstance may those languages be defined as “primitive” — cf. Sapir (1921); changes in language cannot be scientifically considered an “evolution.”

Nevertheless, narrative structures are not identical to linguistic ones: what assures us that the symbols used in the theory do not vary during times and due to cultural changes? Propp first raised this hypothesis: according to him, narrative structures change over time, growing in complexity, and he consequently proposed some univocal criteria to date the different variants of a tale — Propp (1928b) and (1948). His views were close to Goethe’s and Humboldt’s morphogenetic elucidations, which also influenced the work of Saussure — cf. Cassirer (1945).

In my opinion, recursiveness could be an argument in favor of Propp’s intuition. As was previously stated, recursiveness is absent in the corpus of folktale analyzed by Propp. Only some traces of recursive structures may be found in that corpus. Recursiveness may be a modern acquisition in literature. In order to find the first not-trivial conscious attempt at recursive structures, we could focus our

---

12. On this point Galofaro (2012b) raises different objections to explain the crisis of the cognitive paradigm.

research on reported speech, narrations of narrations, and Chinese-box structures. The invention of writing, which allows us to go beyond the limitations of human memory, and the coding of a large number of oral traditions, may have been the necessary condition to develop these techniques: take, for example, the Odyssey, or the Arabian nights.\footnote{Naturally, the historical development of narrative structures can’t be considered a “cultural evolution”, in the Darwinian sense. The identification of literary success with “environmental selection” would only be a vague metaphor; furthermore, there is no casual change in culture. If there is a scientific answer to the question of how culture changes, it implies a theory regarding the formation of the form, which makes use of concepts such as metastability and individuation: a morphogenetic perspective.}

I’m conscious that the techniques of formalization which have been employed in order to formulate the grammar don’t belong to the morphogenetic tradition. Such an algebraic formulation of the theory has always raised many doubts — cf. also Petitot (1985). In my opinion, it cannot represent the dimension of cultural change: the grammar essentially generates all — and only - the symbols we have decided to mechanically implement in the initial set of words and in the rules. If the set of symbols changes, then we have to modify the grammar. Fortunately, a change in the complexity of grammar does not imply that it is impossible to generate the old structures: finite-state narrative structures may be generated through context-free syntagmatic grammars (not vice-versa).

The genesis of new narrative functions and symbols brings to mind Gödel’s main argument against the hypothesis that our mind works as a Turing Machine, which is also an argument against innatism: according to him, our mind could simply produce new sets of symbols, when necessary — cf. Wang (1974). For this reason, we consider a particular generative transformational grammar only as model of a synchronically located competence, which could be re-designed in various ways depending on culture, geographical region and historical period. For these same reasons, a complete semiotic framework involves the proposal of morphogenetic formal models on diachronic cultural change.

\textbf{Bibliographical references}


