Toward a Transcendental Model-Theoretic Semantics for Scientific Languages[*]

Benito MÜLLER[**]

Summary

Based on an idea of *Ajdukiewicz*, a *method of equifunctionality* is developed to provide a formal explication of the notion of *sameness of use* relative to some system of rules. Given this, a set-theoretic explication of *Lauener's* context dependent conception of *synonymy* is introduced by looking at languages of propositional logic, and compared both with Ajdukiewicz's original conception and with *Carnap*'s explication of synonymy based on his *method of extension and intension.*

1. Introduction

1.1. Henri Lauener's Transcendental Theory of Language

The work presented here is part of a larger project I have been engaged in together with Henri Lauener with the aim of formalising his transcendental theory of language. Lauener's views on language,[1] determined by his general philosophical position, i.e. his "open transcendentalism", are rooted in the conviction that semantic and epistemological notions – such as truth, meaning, synonymy, antinomy, analyticity, objectivity and reality – must be thoroughly relativised to what he calls "contexts".[2] These contexts, or "L-*contexts*" (as I shall call them to avoid confusion with other technical uses of the

[*] I wish to thank first and foremost Henri Lauener, not only for having asked me to collaborate with him, but also for the many stimulating and congenial hours we spent discussing in the *Restaurant du Raisin* in Vevey. My thanks also go to Karel Lambert and Hugh Miller for their helpful comments and to the *Schweizerischer Nationalfonds* for having funded the research for this paper.

[**] Wolfson College, Oxford, GB.

[1] See Lauener [1982] - [1992].

[2] Lauener is in particular opposed to the "global holist" conception which views language as a single universal communicative medium where everything is dependent on everything else.

term), are language systems which Lauener conceives of as *theoriebedingte Handlungszusammenhänge,* given in terms of a system of rules governing the use of expressions, and used as instruments for different purposes (such as that of formulating a particular scientific theory). He sees linguistic expressions (i.e. words, phrases, sentences) as instruments, or tools, used according to the system of rules of the L-context adopted for the particular purpose in question. The linguistic characteristics of these expressions are only meant to be determined, or well-defined, relative to such an L-context. This means, in particular, that "meanings", far from being immutable Platonic objects mysteriously related to linguistic individuals, are characteristics which linguistic expressions acquire only when used in accordance with the rules of the L-context in question.[3]

Given the central role of these L-contexts in Lauener's theory of language, the first task in any formalisation of this theory must be to find an adequate formal representation for the involved rule systems. In this essay, I shall focus on a certain kind of rules which feature prominently in L-contexts, namely the kind of rules which guide the "construction" (or "formation") of linguistic expressions and their interpretations. As we shall see, these formation rules allow for a straightforward set-theoretical representation. My aim will be to define a formal representation of Lauener's intuitive context-dependent conception of synonymy within an set-theoretical framework. To set the scene for such a representation, let us return to what is, to my knowledge, the first attempt at giving a formal account of a relativised conception of synonymy, namely

1.2. Ajdukiewicz's Radical Conventionalist Approach[4]

In his 1934 paper entitled *Sprache und Sinn,* Ajdukiewicz gives a detailed account of his radical conventionalist theory of language. Most importantly, for our purposes, he suggests a formal representation of what he calls *meaning rules* of languages. On the basis of this representation, he introduces a formal method for establishing a relation of "*isotopy*" between expressions (of a given language), a relation which he then identifies with that of synonymy. Ajdukiewicz's position on this subject matter is admirably summarised in the following passage by Giedymin:

[3] To ask about the meaning of an expression without specifying an L-context is thus on a par to inquiring about the numerical weight of an object without specifying a measurement system.

[4] See Ajdukiewicz [1934a], [1934b], and [1935].

"[In Ajdukiewicz's abstract analysis] language is reconstructed in terms of its vocabulary, the rules of syntax and the *meaning-specification* or *meaning-* (acceptance) *rules.* The latter determine the *structure* or *matrix of the language* and the *meaning* of expressions is then conceived as an abstract property they have in virtue of the *positions* they take *in the matrix.* Three kinds of meaning-rules may be distinguished as basic: (1) *axiomatic meaning-rules* specify sentences which are to be accepted unconditionally: the rejection of any sentence dictated by an axiomatic meaning-rule amounts to the violation of the meaning-specification characteristic for the language; (2) *deductive meaning-rules* specify ordered pairs of sentences (or ordered pairs whose first element is a sentence-class and the second element is a sentence) such that if one accepts the first of them one is thereby committed to accepting the second on pain of violating the meaning-specification of the language; (3) *empirical meaning-rules* assign to definite experiental data sentences (simple empirical meaning-rules) or to definite experiential data and sentences, other sentences (compound empirical meaning-rules) such that in the presence of those data (possibly conjoined with the acceptance of some sentences) one is forced to accept the co-ordinated sentence if one is to avoid violation of meaning. Axiomatic and deductive meaning-rules are *discursive,* they are sufficient for purely discursive languages, e.g. of pure mathematics." [Giedymin 1978, p. XXXVII]

1.2.1. Language Matrices

Ajdukiewicz's formal representation of a system (say $\mathcal{M}_A$) of meaning rules as a matrix ($\mathbf{M}_A$), is as follows:

"Each meaning rule has a *scope:* the scope of an axiomatic meaning-rule is a set of sentences (the axioms or principles of the language); the scope of a deductive meaning-rule is a set of ordered pairs of sentences (or of sentence-class/sentence); the scope of an empirical meaning-rule is a set of ordered pairs of experiential data and sentences or experiential data-*cum*-sentences/sentence). The scopes of the meaning-rules of the same type may be summed. The sum of the scopes of all meaning-rules of the same type is their *total scope.*" [Giedymin, 1978, p. XXXVII]

The total scopes for the rules in $\mathcal{M}_A$ (i.e., if we wish, the scope of $\mathcal{M}_A$) are then re-described in matrix form with the additional modification of representing the syntactic structure of a sentence, say '$(p \vee q) \leftrightarrow (\neg p \rightarrow q)$', by "encoding" it in a sequence of expressions, in our case

$$\langle (p \vee q) \leftrightarrow (\neg p \rightarrow q), \leftrightarrow, (p \vee q), \vee, p, q, (\neg p \rightarrow q), \rightarrow, \neg p, \neg, p, q \rangle.$$

The example of a language matrix given by Ajdukiewicz [1934a, p. 131] is for a language with the following finite set of expressions $\{a,b,c,d,e,f,g,h,i,j,k\}$. The matrix contains three parts, each of which representing (the scope of) one of the three kinds of "meaning rules" accepted by Ajdukiewicz, namely

$$\text{Axiomatic Part} \quad \left\{ \begin{array}{l} \langle a,b,c \rangle \\ \langle d,k,a,b,c,e,f,g \rangle \end{array} \right.$$

$$\text{Deductive Part}\qquad \begin{cases} \langle a,b,c\rangle, \langle e,f,g\rangle \\ \langle d,k,a,d,c,e,f,g\rangle, \langle i,j,b\rangle \\ \langle e,f,g\rangle, \langle h\rangle \end{cases}$$

$$\text{Empirical Part}\qquad \begin{cases} \alpha,\ \langle h\rangle \\ \beta,\ \langle e,f,g\rangle \\ \gamma,\ \langle h\rangle \end{cases}$$

where α,β, and γ represent "*Erfahrungsdaten*", i.e. experiential data.

On the basis of these matrix representations of meaning-rules, Ajdukiewicz then gives a formally precise definition of a relation, say $\equiv_A$, which he puts forward as an explication of synonymy:

> "Zwei Ausdrücke einer Sprache nennen wir synonym, wenn sie in der Matrix der Sprache isotop sind, d.h. wenn die Matrix bis auf die Ordnung der Zeilen unverändert bleibt, falls man in ihr beide Ausdrücke vertauscht." [Ajdukiewicz 1934a, p. 132]

That is, two expressions are called synonymous (by Ajdukiewicz) if they are isotopes in the matrix based on the total scopes of the meaning rules.

1.2.2. Tarski's Objection

It is one thing simply to call a formally defined relation "synonymy", and quite another whether the relation so called actually represents the intuitive relation of that name. Indeed

> "it was pointed out by Tarski (as reported by Ajdukiewicz in 'The Problem of Empiricism and the Concept of Meaning') that, at least for a language *L* based exclusively on axiomatic and deductive meaning rules, one can construct two expressions *A* and *B* such that the meaning rules of *L* are invariant under the exchange of *A* and *B* and yet *A* and *B* have *non*-identical denotations." [Giedymin 1978, p. XLVII]

To give an example: in a purely discursive language with $\neg(t = f)$ and $\neg(f = t)$ as (sole) axiomatic meaning rules, we will have that $t \equiv_A f$. Yet clearly the denotations of *t* and *f* must differ in any model of the two axioms. Thus, if we adopt the view that synonymy entails identity of denotations, then $\equiv_A$ cannot be the relation of synonymy.

Giedymin [1978, p. XLVII] and Ernest [1985, p. 228] suggest that, in order to solve this problem, all we need to do is add the sameness of denotation as a further criterion to Ajdukiewicz's definition of synonymy.[5] It might

[5] Thus Ernest, in his analysis of mathematical languages, adopts the view that two expressions Φ_1 and Φ_2 of a language *L* are synonymous (relative to a set of postulates *Π* and the (Fortsetzung siehe nächste Seite)

well be – in particular for the mathematical languages considered by Ernest – that this added requirement ensures that the relation so defined is co-extensive with that of synonymy. Yet even if it is, I believe that the way in which this is achieved is too *ad hoc* to reflect the complex interdependencies of semantic and syntactic features involved in synonymy. I shall return to his issue briefly towards the end of this essay.

1.3. Equifunctionality

Tarski's objection shows that Ajdukiewicz's isotopy-relation cannot be (a formal representation of) synonymy. And yet, in defining this relation, Ajdukiewicz has given us an insight which, in my view, far outweighs this shortcoming. He has brought to the fore the notion of isotopy relative to the scope of a rule system, and he has given us a general method of how to establish such isotopy relations. In the analysis to follow, I shall apply this method to different kinds of rule systems, with the difference that – instead of interpreting the resulting isotopy relations as *meaning* relations – I shall generally interpret them as relations of having the *same use* (or *function*) relative to the rule system in question. For the purposes of this introductory essay, I propose to use an L-context L_0 of a particularly simple type – namely that of *propositional L-contexts* used to talk about propositional logic – in discussing the different kinds of (formation) rule systems and in explicating the different kinds of "sameness-of-use", or "equifunctionality" relations given relative to these rule systems. The idea being, of course, that for some such rule system, *sameness of use* will indeed be *sameness of meaning* (in L_0). To do this, let me introduce the following definitions.

(i) A *system of formation rules,* as I shall use the term, is a system $\mathcal{R}$ of rules associated with a unique and well-defined class – **Scope**($\mathcal{R}$) – of objects referred to as the *scope of* $\mathcal{R}$. Paradigmatically this scope will be the class of all the objects which can be *constructed* (or *formed*) by means of $\mathcal{R}$.

Thus, if $\mathcal{R}$ is a system of grammatical rules, **Scope**($\mathcal{R}$) will be the class of all the expressions that can be correctly formed according to $\mathcal{R}$; or, if it is the system of rules governing a certain board-game, say chess, then its scope will be the class of all the sequences of configurations on the chessboard which

(Fortsetzung von S. 4)
class M of *L*-structures satisfying *Π*) if, and only if, for all M ∈ M: $\Phi_1 \equiv_A \Phi_2$ (relative to *Π*) *and* **ext** (Φ_1, M) = **ext** (Φ_2, M) – where **ext** (Φ_i, M) designates the extension/denotation of Φ_i in M.

constitute correctly played games of chess. Crucial in this conception is that the scope of such a system reflects its numerical identity[6] – i.e. that for any formation rule systems $\mathcal{R}$ and $\mathcal{R}'$

$$\mathcal{R} = \mathcal{R}' \text{ iff } \mathbf{Scope}(\mathcal{R}) = \mathbf{Scope}(\mathcal{R}')$$

– for this enables us to use the scope of a rule system as its set-theoretic representation. Given this, we can now introduce the following terminology concerning a system of formation rules $\mathcal{R}$ and a collection **C** of components used in $\mathcal{R}$-constructions:

(ii) A set $\mathbf{A} \supseteq \mathbf{Scope}(\mathcal{R})$ is and *Ajdukiewicz-field* (*for* $\mathcal{R}$, *relative to* **C**) if for every $X \in \mathbf{A}$ and every $a,b \in \mathbf{C}$ an *exchange* of a and b in X is well-defined and produces a unique result, say $X\|^a_b$, which lies in **A**.

(iii) The mapping $\|^a_b : \mathbf{A} \xrightarrow[onto]{1\text{-}1} \mathbf{A}; X \mapsto X\|^a_b$ induced by any given $a,b \in \mathbf{C}$ in such an Ajdukiewicz-field shall be referred to as the *Ajdukiewicz-automorphism* (induced by a and b).[7]

(iv) $\mathcal{R}$ is an *Ajdukiewicz-system* (relative to **C**) if there is an Ajdukiewicz-field for $\mathcal{R}$ (relative to **C**).

Assuming a certain compatibility between the "exchange constructions" associated with Ajdukiewicz-fields of such an Ajdukiewicz-system $\mathcal{R}$,[8] we can then define the following formal equivalence relation

(v) $$a \overset{\mathcal{R}}{\equiv} b \text{ iff}_{\text{df}} \mathbf{Scope}(\mathcal{R})\|^a_b = \mathbf{Scope}(\mathcal{R})$$

[6] Even though we might well be able, in certain cases, to judge the numerical sameness or difference of a formation rule system $\mathcal{R}$and a formation rules system $\mathcal{R}'$on the basis of their procedural characterisations alone (i.e. without having to compare scopes explicitly), this only reflects the fact that in these cases the relevant scope comparisons can be made implicitly. The numerical individuation of formation rule systems by their scopes is essential to the conception of such systems.

[7] The fact that these mappings are automorphisms, i.e. one-one and onto, follows directly from certain intuitive constraints which $X\|^a_b$ must satisfy, in order to be product of an "exchange constructions", namely that

$$X\|^a_a = X, \quad X\|^a_b = X\|^b_a \text{ and } (X\|^a_b)\|^a_b = X.$$

[8] The assumption here is that an exchange, of a and b in some object X is determined exclusively by the structure of X, meaning that it is *not* relative to the Ajdukiewicz-fields which X happens to be a member of. Or, to put it differently, if **A** and **A**′ are both Ajdukiewicz-fields for $\mathcal{R}$with the Ajdukiewicz-automorphisms ${}_{\mathbf{A}}\|^a_b$ and ${}_{\mathbf{A}'}\|^a_b$ respectively, then

$${}_{\mathbf{A}}\|^a_b\,|_{\mathbf{A}\cap\mathbf{A}'} = {}_{\mathbf{A}'}\|^a_b\,|_{\mathbf{A}\cap\mathbf{A}'}.$$

It is this assumption which allows us to omit a reference to the relevant Ajdukiewicz-field in the notation for the Ajukiewicz-automomorphisms, i.e. to use $\|^a_b$ instead of the more cumbersome ${}_{\mathbf{A}}\|^a_b$.

which is independent of the choice of Ajdukiewicz-field, and which I shall refer to as the *equifunctionality* relation of $\mathcal{R}$.[9]

It is this relation which I shall use as representation of the informal relation of *having the same use* or *function* as specified by the rule system $\mathcal{R}$. The justification for this is based on the fact certain construction rule systems allow for a special kind of procedural definition which enables us to give an informal characterisation of what "sameness of use" (relative to such a system) amounts to. What I have in mind are systems in which each of the basic components in question is referred to in precisely one of the rules. Take, for example, the system $\mathcal{S}^{\omega}$ for constructing number-sequences by means of the numerals 1, 2, and 3, given by

S_1 : $\langle n \rangle \in$ **Scope** $(\mathcal{S}^{\omega})$ iff $n = 1$;
S_2 : if $s, s' \in$ **Scope** $(\mathcal{S}^{\omega})$ then $s^{\frown}\langle 2 \rangle^{\frown} s' \in$ **Scope** $(\mathcal{S}^{\omega})$;
S_3 : if $s, s' \in$ **Scope** $(\mathcal{S}^{\omega})$ then $s^{\frown}\langle 3 \rangle^{\frown} s' \in$ **Scope** $(\mathcal{S}^{\omega})$.

The "uses" of the basic components 1, 2, and 3 relative to $\mathcal{S}^{\omega}$ are fixed by the procedural instructions given in S_1, S_2 and S_3 respectively, and nothing else. Thus if, by an exchange of reference to the basic components in question, two such rules are mutually translated into one-another – as is the case for S_2 and S_3 – then these basic components must have the same use relative to the system in question. Given that, in this case, the system specified by the "translated" rules is numerically the same (i.e. has the same scope) as $\mathcal{S}^{\omega}$, it is easy to see, from the way in which our equifunctionality relations were defined, why they can be used to represent the relevant sameness of use (an analogous argument can be given for difference of use). As for other rule systems, where we do not have the same sort of intuitive grasp of the sameness of use, the equifunctionality relations are assumed to have the desired representational character as a matter of convention.

2. *Syntactic Formation Rules*

Let us then begin to discuss specifically the formation rule systems involved in propositional L-contexts. Amongst these rule systems there is one – namely the system $\mathcal{G}$ of syntactic ("*grammatical*") formation rules – which provides a natural starting point for our investigation. To simplify matters fur-

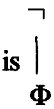
[9] In the terms given in these definitions, Ajdukiewicz's isotopy-relation can now be characterised as $x \,|_A\, y$ iff **Scope** $(\mathcal{M}_A) \|^{x}_{y} =$ **Scope** $(\mathcal{M}_A)$ (for $x, y \in \{a, \ldots, k\}$) – with **Scope** $(\mathcal{M}_A)$ being the set of the three part of $\mathbf{M}_A$, themselves treated as sets (which simply takes into account Ajdukiewicz's use of *isotopy* instead of identity).

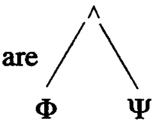
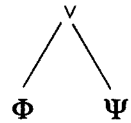
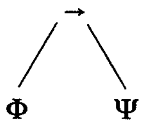
ther, let me focus my analysis on the classical propositional L-context $\boldsymbol{L}_0$ based on the set $\boldsymbol{B} := \{\neg, \wedge, \vee, \rightarrow, p_1, p_2, p_3, \ldots\}$ as *basic linguistic expressions.*[10]

2.1. The Tree-Structure Interpretation of $\mathcal{G}$

The "procedural" characterisation of $\mathcal{G}$ – that is the description of $\mathcal{G}$ as a collection of rules, as opposed to its set-theoretical characterisation in terms of its scope – which I shall adopt, is given in the following recursive formation rules for certain tree-structures, which I shall refer to as (*syntactically*) *well-formed formulae* (*wffs*):[11]

i) A one-node tree-structure $\langle x \rangle$ is a wff iff $x \in \boldsymbol{Var}$;

ii) If Φ is a wff, then so is 

iii) If Φ and Ψ are wffs, then so are   and 

– where $\boldsymbol{Var} = \{p_1, p_2, p_3, \ldots\}$ is the set of "propositional variables". From the way in which these rules are stated, we can – like in the case of $\mathcal{S}^{\omega}$ – intuitively gauge the sort of functions assumed by vocabulary expressions in virtue of $\mathcal{G}$, namely the functions of being a *variable*, a *monadic logical connective* and a *dyadic logical connective.* That is to say, the informal "sameness of use structure" imposed on $\boldsymbol{B}$ by $\mathcal{G}$ is given by

$$\boldsymbol{B}^{*} = \{\{\neg\}, \{\wedge, \vee, \rightarrow\}, \boldsymbol{Var}\}$$

– where $\{\neg\}$ is the set of monadic logical connectives, and $\{\wedge, \vee, \rightarrow\}$ that of the dyadic ones.

[10] More traditionally $\boldsymbol{B}$ is referred to as the "vocabulary" of this L-context.

[11] The choice of interpreting $\mathcal{G}$ as a system for the construction of tree-structures (out of the basic linguistic expressions of $\boldsymbol{L}_0$ given in $\boldsymbol{B}$) is by no means the only possible one. Thus one might interpret $\mathcal{G}$ as a system for the construction of sequences of vocabulary elements (which could be called the "purely sequential interpretation"), but this interpretation would, at least if based on the traditional notation, have to be rejected because of its inability to represent certain intuitive grammatical distinctions: no distinction could, for example, be made between '$(p_1 \wedge p_2) \vee p_3$' and '$p_1 \wedge (p_2 \vee p_3)$'. Of course there are notational systems where the additional structure is encoded sequentially (see, for example, the notation used in Bell and Machover [1977]), but that does not mean that it is not inherent in the constructed objects, but merely that we have additional rules to make it explicit. The advantage of interpreting $\mathcal{G}$ as a system of rules for constructing tree-structures is that these tree-structures incorporate precisely the sort of "sameness" Carnap refers to on pp. 57 f. in his [1956].

The class $\mathbf{F}_0$ of all finitely branched, finite trees based on $\boldsymbol{B}$ is an Ajdukiewicz-field for $\mathcal{G}$ relative to $\boldsymbol{B}$.

To show this we need to establish a more precise conception of an exchange of basic L_0-expressions in the tree-structures of $\mathbf{F}_0$. My suggestion is to explicate this in terms of the formal notion of substitution. Take, for example, the wff $s =$ '$(p_1 \vee p_2) \vee p_3$'. It is intuitively clear that the transformation of s described as "the exchange of 'p_1' and 'p_2' in s" has

$$s\|^{p_2}_{p_1} = \text{‘}(p_2 \vee p_1) \vee p_3\text{’}$$

as its unique product, and that this product is again an element of $\mathbf{F}_0$. The question is whether this transformation can be described in terms of substitutions. At first sight, one might consider an explication of $\Phi \|^a_b$ *(for some a,b* $\in \boldsymbol{B}$ and $\Phi \in \mathbf{F}_0$) as $\Phi|^{a,b}_{b,a}$ – where $X|^a_b$ designates the product of substituting a for b in X, and $X|^{a,b}_{b,a}$ is short for $(X|^a_b)|^b_a$. The fact, however, that

$$s|^{p_1,\, p_2}_{p_2,\, p_1} = \text{‘}(p_1 \vee p_1) \vee p_3\text{’},$$

and

$$s|^{p_2,\, p_1}_{p_1,\, p_2} = \text{‘}(p_2 \vee p_2) \vee p_3\text{’}$$

shows that this explication cannot be correct: not only because neither of the substitutions results in the intuitive exchange product, i.e. '$(p_2 \vee p_1) \vee p_3$', but also because of the fact that since $s|^{p_1,\, p_2}_{p_2,\, p_1} \neq s|^{p_2,\, p_1}_{p_1,\, p_2}$, the suggested explication allows for an a-symmetry not admissible for exchange operations.[12] The simplest way of overcoming this obstacle is by employing "place holders" – by which I mean characters (say ξ and ζ) which are not part of the vocabulary – in intermediary substitutions, i.e. to characterise $\Phi \|^a_b$ as $(\Phi|^{a,b}_{\xi,\zeta})|^{\xi,\zeta}_{b,a}$. It is easy to see that this substitutional characterisation avoids the shortcomings of the initial description,[13] and that

$$\Phi \|^a_b := (\Phi|^{a,b}_{\xi,\zeta})|^{\xi,\zeta}_{b,a}$$

defines an automorphism on $\mathbf{F}_0$ for any $a,b \in \boldsymbol{B}$. The "*syntactic*" equifunctionality relation $\overset{\mathcal{G}}{\equiv}$ (on $\boldsymbol{B}$) is then given by

$$a \overset{\mathcal{G}}{\equiv} b \text{ iff}_{\text{df}} \textbf{ Scope}(\mathcal{G}) \|^a_b = \textbf{Scope}(\mathcal{G}) \text{ (for } a,b \in \boldsymbol{B}\text{)},$$

[12] The product of exchanging a and b in Φ cannot be different from the product of exchanging b and a in Φ, i.e. $\Phi\|^a_b$ must be the same as $\Phi\|^b_a$.

[13] Not only do we have that $(\Phi|^{a,b}_{\xi,\zeta})|^{\zeta,\,\xi}_{b,\,a} = (\Phi|^{b,a}_{\zeta,\xi})|^{\zeta,\xi}_{a,b}$, but also, as desired, that

$$(\text{‘}(p_1 \vee p_2) \vee p_3\text{’}|^{p_1,p_2}_{\xi,\,\zeta})|^{\xi,\,\zeta}_{p_2,p_1} = \text{‘}(p_2 \vee p_1) \vee p_3\text{’}$$

and we find that the $\overset{\mathcal{G}}{\equiv}$-equivalence classes coincide with our informal classification of the basic $\boldsymbol{L}_0$-expressions, i.e. that

$$\boldsymbol{B}\big|_{\overset{\mathcal{G}}{\equiv}} = \boldsymbol{B}^{*}.$$

The syntactic equifunctionality relation $\mathcal{G}$ thus groups together precisely those vocabulary expressions which we informally said to have the same function relative to the adopted grammatical rule system. Our "*equifunctionality method*" has thus yielded a set-theoretical characterisation of the relation (between vocabulary expressions) of having the same function relative to $\mathcal{G}$. But what about (complex) well-formed formulae? Is it possible to give the same sort of set-theoretical characterisation of the sameness-of-$\mathcal{G}$-function as far as they are concerned?

2.2. The $\mathcal{G}$-Equifunctionality of Well-Formed Formulae

The problem with extending $\mathcal{G}$ to well-formed formulae is the following: since well-formed formulae can be proper parts of one another, it is no longer self-evident how their exchange in a given construct is to be executed. Take again $s =$ '$(p_1 \vee p_2) \vee p_3$', this time as an example of a $\mathcal{G}$-construct within which an exchange of two constituent wffs is to be executed. There is no problem if the two wffs are disjoint – like $\Phi =$ '$p_1 \vee p_2$' and $\Phi' =$ 'p_3': in this case the exchange is well-defined and its product is $s\|^{\Phi}_{\Phi} =$ '$p_3 \vee (p_1 \vee p_2)$'. But what are we to understand by an exchange of say $\Phi =$ '$p_1 \vee p_2$' and $\Phi'' =$ 'p_1' in s? The problematic nature of this becomes clear if we try to apply our substitutional characterisation of such exchanges, for we find that

$$(s|^{\Phi,\Phi''}_{\xi,\,\zeta})\,|^{\xi,\ \ \zeta}_{\Phi'',\Phi} = \text{'}p_1 \vee p_3\text{'}, \text{ and } (s|^{\Phi'',\Phi}_{\zeta,\ \xi})\,|^{\zeta,\,\xi}_{\Phi,\Phi''} = \text{'}((p_1 \vee p_2) \vee p_2) \vee p_3\text{'},$$

a result which – assuming that the employed substitutional explication is correct – shows that the notion of an exchange is ill-defined in the present example.[14] To overcome this obstacle, I propose to use a "translation" of the rule system in question (i.e. in the present case, a "translation" of $\mathcal{G}$) in which the complex expressions to be exchanged (i.e. Φ and Φ'') are "translated" into atomic vocabulary expressions. More precisely, for any given "exchange expressions" $\Psi,\Psi' \in$ **Scope**$(\mathcal{G})$, an auxiliary rule system, say $\mathcal{G}^{\xi\zeta}$, is introduced as a sub-set of the class $\mathbf{F}_0^{\xi\zeta}$ of all finitely branched, finite trees based on $\boldsymbol{B}^{\xi\zeta} = \boldsymbol{B} \,\dot{\cup}\, \{\,\xi,\zeta\,\}$ by defining

[14] The idea being that if an *exchange* of x and y in X is well-defined then

$$X\|^{x}_{y} = (X|^{x,\,y}_{\xi,\zeta})|^{\xi,\zeta}_{y,x} = (X|^{x,\ y}_{\xi,\ \zeta})|^{\zeta,\xi}_{x,\ y}.$$

$$\mathbf{Scope}(\mathcal{G}^{\xi\zeta}) := \mathbf{Scope}(\mathcal{G})\,\big|^{\Psi,\,\Psi'}_{\xi,\,\zeta} \cup \mathbf{Scope}(\mathcal{G})\,\big|^{\Psi',\Psi}_{\zeta,\;\xi}\,.$$

The $\mathcal{G}$-functions of Ψ and Ψ' are then "mirrored" in the $\mathcal{G}^{\xi\zeta}$-functions of ξ and ζ, respectively. Given that $\mathbf{Scope}(\mathcal{G}^{\xi\zeta})$ is nothing but the class of well-formed formulae of the propositional language, say $\boldsymbol{L}_0^{\xi\zeta}$, one obtains by adding ξ and ζ as new *propositional variables* to our vocabulary $\boldsymbol{B}$, and that $\mathbf{F}_0^{\xi\zeta}$ is an Ajdukiewicz-field relative to this enriched vocabulary $\boldsymbol{B}^{\xi\zeta}$, we are thus able to compare the $\mathcal{G}$-functions of Ψ and Ψ' by looking at whether their "proxies" (i.e. ξ and ζ) are $\mathcal{G}^{\xi\zeta}$-equifunctional or not. In other words, we can use the unproblematic Ajdukiewicz-automorphism on $\mathbf{F}_0^{\xi\zeta}$ induced by ξ and ζ, i.e.

$$\|^{\xi}_{\zeta}\colon \mathbf{F}_0^{\xi\zeta} \xrightarrow[\textit{onto}]{\textit{one-one}} \mathbf{F}_0^{\xi\zeta},$$
$$\Phi \mapsto \Phi\|^{\xi}_{\zeta}$$

to specify the *syntactic* ($\mathcal{G}$-) equifunctionality of Ψ and Ψ' as

$$\Psi \overset{\mathcal{G}}{\doteqdot} \Psi' \ \text{iff}_{\text{df}}\ \xi \overset{\mathcal{G}^{\xi\zeta}}{\doteqdot} \zeta$$

– with $\xi \overset{\mathcal{G}^{\xi\zeta}}{\doteqdot} \zeta$ iff$_{\text{df}}$ $\mathbf{Scope}(\mathcal{G}^{\xi\zeta})\|^{\xi}_{\zeta} = \mathbf{Scope}(\mathcal{G}^{\xi\zeta})$. And we find – not surprisingly, since all propositional variables of $\boldsymbol{L}_0^{\xi\zeta}$ are $\mathcal{G}^{\xi\zeta}$-equifunctional – that the scope of $\mathcal{G}$ forms one single $\overset{\mathcal{G}}{\doteqdot}$-equivalence (i.e. syntactic equifunctionality) class:

$$\mathbf{Scope}\,(\mathcal{G})\,\big|_{\overset{\mathcal{G}}{\doteqdot}} = \{\mathbf{Scope}\,(\mathcal{G})\},$$

i.e. that *all* well-formed $\mathcal{G}$-formulae have the same function (or are used in the same way) relative to $\mathcal{G}$.

3. Semantic Rules

With this set-theoretical characterisation of the purely syntactic functions of linguistic expressions, we can now turn to certain functions which these expressions have over and above the purely syntactic ones, functions which they attain in virtue of being used according to what I shall generically refer to as *semantic rules.*

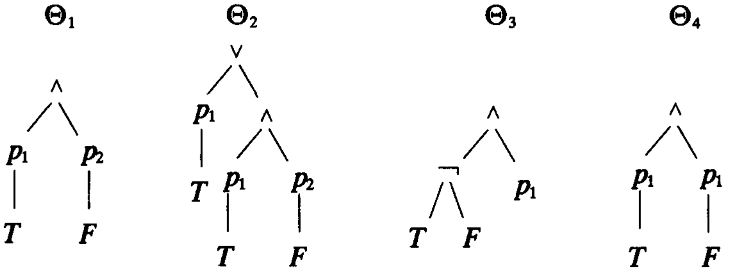
3.1. Semantic Formation Rules

Semantic rules are rules which determine the proper way of adding extra ("semantic") features to the syntactic constructs we have so far obtained. They accordingly transform our $\mathcal{G}$-constructs into a species of what might be called "semantic individuals". Let us begin our discussion of these rules by looking at the system $\mathcal{S}$ of (classical) *semantic formation rules* for our propositional L-context which determines the proper way of assigning denotations to the propositional variables of wffs. I shall assume that these denotations are *truth-values*, which shall collectively be given by the set **Val** $= \{T, F\}$. The question is, how exactly are we to represent these "semantically enriched" well-formed formulae?

Given the adopted interpretation of $\mathcal{G}$ as a system governing the formation of (syntactically well-formed) tree-structures, the most natural interpretation of the envisaged structural enrichment seems to be that of an addition of extra ("denotation") branches by which syntactically well-formed formulae are transformed into structurally enriched tree-structures – which I shall refer to as *semantically well-formed formulae* or *$\mathcal{S}$-formulae* – according to the following rule:

> To transform a $\mathcal{G}$-formula Φ into a $\mathcal{S}$-formula Θ is to add *one* (and only one) *simple truth-value branch* – i.e. a branch whose only node is occupied by a truth-value – to every node of Φ occupied by a variable in a *syntagmatically homogeneous* manner, that is in such a manner that if $p, p' \in$ *Var* occur in Θ, and v, v' are the truth-values of the terminal nodes attached to p and p', respectively, then $p = p'$ entails $v = v'$.

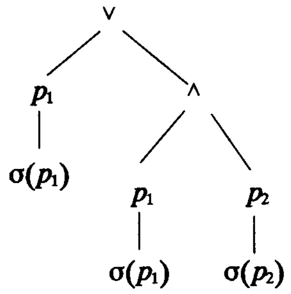
$\mathcal{S}$ is thus conceived as a system of rules for the formation of a certain kind of finitely branched, finite tree-structures (based on $B \dot{\cup}$ **Val**), and it is easy to see that only the first two of the following trees of this type



are semantically well-formed in this sense.[15] These semantically well-formed formulae are closely related to the more traditional *interpreted formula*, them-

[15] Θ_3 is semantically ill-formed, not only on account of it not being based on a syntactically well-formed formula, but also became it contains, on the one hand, a node with more than one truth-value assignment, and, on the other, a variable-node with none. Θ_4 is ill-formed because it does not satisfy the syntagmatic homogeneity requirement: one and the same denoting symbol (namely p_1) with a multiple occurrence in the underlying syntagmatic unit (i.e. '$p_1 \wedge p_1$') is assigned different denotations, namely T in one occurrence, and F in the other.

selves given by a well-formed formula (in the traditional sense) and an *interpretation* of the denoting symbols of the language in question, which, in the case of L_0, is usually referred to as an assignment $\sigma : \boldsymbol{Var} \to \mathbf{Val}$ of truth-values to the propositional variables. Indeed, one can easily specify semantically well-formed formulae in terms of such assignments; for every semantically well-formed formula Θ there is a $\mathcal{G}$-formula Φ and an assignment $\sigma \in \Sigma$ – where Σ denotes the class of all the truth-value assignments relevant for L_0 – such that the tree-structure, say $k\langle \Phi, \sigma \rangle$, obtained by adding a simple $\sigma(p_i)$-branch to every p_i-node of Φ is nothing else but Θ, i.e. such that $k\langle \Phi, \sigma \rangle = \Theta$. Take, for example, the semantically well-formed formula Θ_2 illustrated above: $k\langle p_1 \vee (p_1 \wedge p_2), \sigma \rangle$, that is



is clearly the same as Θ_2, provided we choose an assignment with $\sigma(p_1) = T$ and $\sigma(p_2) = F$. We have thus established a functional relationship, say again k, between $\mathcal{G}$-formulae and truth-value assignments, on the one hand, and $\mathcal{S}$-formulae, on the other, which is many-one and onto – namely

$$k\colon \boldsymbol{Scope}(\mathcal{G}) \times \Sigma \xrightarrow[onto]{many\text{-}one} \boldsymbol{Scope}(\mathcal{S})$$

$$\langle \Phi, \sigma \rangle \mapsto k\langle \Phi, \sigma \rangle$$

– and it is easy to see that the construction employed here (and hence the function k) can be extended to the super-set $F_0 \times \Sigma$ of $\boldsymbol{Scope}(\mathcal{G}) \times \Sigma$.[16] With this, we can now give a simple characterisation of an exchange of basic expressions, say a and b, not only in semantically well-formed formulae, but in any of the products of the extended k-construction – i.e. in any $X \in F_1 := k[F_0 \times \Sigma] \supset \boldsymbol{Scope}(\mathcal{S})$ – namely

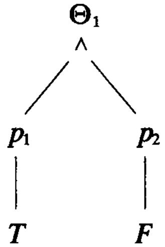
[16] All that needs to be done to avoid possible construction ambiguities is to add the condition that the $\sigma(p_i)$-branches are to be added as the first (or, alternatively, the last) of the branches of the relevant p_i-node, just in case we are dealing with a p_i-node which is not terminal.

$$(*) \qquad X\|^a_b := k\langle \Phi\|^a_b, \sigma\|^a_b \rangle \quad (\text{for } X = k\langle \Phi, \sigma\rangle \in F_1)$$

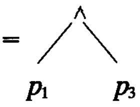
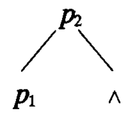
– with $\sigma\|^a_b$: ***Var*** → **Val** given as

$$\sigma\|^a_b(x) = \begin{cases} \sigma(a) & \text{if } x = b \,\&\, a, b \in \mathit{Var}; \\ \sigma(b) & \text{if } x = a \,\&\, a, b \in \mathit{Var}; \\ \sigma(x) & \text{else.} \end{cases}$$

Strictly speaking, we ought to have made a notational distinction between the exchange operation to be defined, and the one given on F_0, say by writing $X_{\mathrm{F}_1}\|^a_b$ and $\Phi_{\mathrm{F}_0}\|^a_b$, respectively. However, to keep the notation less cumbersome, I shall assume these differences as implicitly understood.[17]
Since $\sigma\|^a_a = \sigma$, $\sigma\|^a_b = \sigma\|^b_a$ and $(\sigma\|^a_b)\|^a_b = \sigma$, it follows that $X\|^a_b$ – as defined above – satisfies the general conditions placed on exchange operations.[18]
To exemplify all this, let us look at an exchange of 'p_2' with 'p_3', on the one hand, and with '$\wedge$', on the other, in the first of the two semantically well-formed formulae illustrated above, i.e. in



Assuming $\Theta_1 = k\langle p_1 \wedge p_2, \sigma_1\rangle$ (i.e. that $\sigma_1(p_1) = T$ and $\sigma_1(p_2) = F$) we have that

$(p_1 \wedge p_2)\|^{p_2}_{p_3} =$  $(p_1 \wedge p_2)\|^{p_2}_{\wedge} =$ 

[17] Note, incidentally, that $X\|^a_b$ – as specified in (*) – is only well-defined in virtue of the fact that $k\langle \Phi, \sigma\rangle = k\langle \Phi', \sigma'\rangle$ implies $k\langle \Phi\|^a_b, \sigma\|^a_b\rangle = k\langle \Phi'\|^a_b, \sigma'\|^a_b\rangle$.

[18] Given $X = k\langle \Phi, \sigma\rangle$, we have that

a) $X\|^a_a = k\langle \Phi, \sigma\rangle\|^a_a = k\langle \Phi\|^a_a, \sigma\|^a_a\rangle = k\langle \Phi, \sigma\rangle = X.$

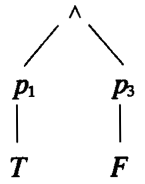
b) $X\|^a_b = k\langle \Phi, \sigma\rangle\|^a_b = k\langle \Phi\|^a_b, \sigma\|^a_b\rangle = k\langle \Phi\|^b_a, \sigma\|^b_a\rangle = k\langle \Phi, \sigma\rangle\|^b_a = X\|^b_a.$

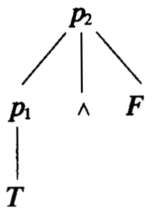
c) $(X\|^a_b)\|^a_b = (k\langle \Phi, \sigma\rangle\|^a_b)\|^a_b = k\langle \Phi\|^a_b, \sigma\|^a_b\rangle\|^a_b = k(\Phi\|^a_b)\|^a_b, (\sigma\|^a_b)\|^a_b\rangle = k\langle \Phi, \sigma\rangle = X.$

and that

$$\sigma_1\|^{p_2}_{p_3}(p_1) = \sigma_1(p_1) = T \qquad \sigma_1\|^{p_2}_{\wedge}(p_1) = \sigma_1(p_1) = T$$
$$\sigma_1\|^{p_2}_{p_3}(p_3) = \sigma_1(p_2) = F \qquad \sigma_1\|^{p_2}_{\wedge}(p_2) = \sigma_1(p_2) = F$$

and consequently we get the following exchange products:

$\Theta_1\|^{p_2}_{p_3} =$ 

$\Theta_1\|^{p_2}_{\wedge} =$ 

The former of the two clearly *is* semantically well-formed, while the latter, equally clearly, is *not*:

$$\Theta_1\|^{p_2}_{p_3} \in \mathit{Scope}(\mathcal{S}) \qquad \Theta_1\|^{p_2}_{\wedge} \notin \mathit{Scope}(\mathcal{S})$$

$\boldsymbol{F}_1$ is an Ajdukiewicz-field for $\mathcal{S}$ relative to $\boldsymbol{B}$, and, given the above examples, it will not be surprising that the "*semantic*" equifunctionality relation $\overset{\mathcal{S}}{\equiv}$ for basic $\boldsymbol{L}_0$-expressions – given, according to our general scheme, by

$$a \overset{\mathcal{S}}{\equiv} b \quad \text{iff}_{\text{df}} \quad \mathit{Scope}(\mathcal{S})\|^{a}_{b} = \mathit{Scope}\,\mathcal{S} \quad (\text{with } a, b \in \boldsymbol{B}) -$$

is *co-extensional* with the syntatic equifunctionality relation, i.e. that

$$a \overset{\mathcal{G}}{\equiv} b \quad \text{iff} \quad a \overset{\mathcal{S}}{\equiv} b \qquad (\text{for } a, b \in \boldsymbol{B}).$$

To illustrate, however, that this need not to be the case, let us briefly look at propositional L-contexts with propositional *constants*, that is the case where a set *Const*: $= \{c_1, \ldots, c_k\}$ is added to our vocabulary $\boldsymbol{B}$ to form the propositional L-context $\boldsymbol{L}_0^{Const}$. As far as the *syntatic* formation rules are concerned, these constants are treated on a par with variables, and thus we will find that

$$\boldsymbol{B}\Big|_{\underset{\equiv}{\mathcal{G}}^{Const}}\{\{\neg\},\{\wedge, \vee, \rightarrow\},\ \mathit{Var} \cup \mathit{Const}\}.$$

The functional differentiation of variables and constants only occurs on the level of semantic formation rules, in that of the two, only *constants* are stipulated to have *one and the same* denotation throughout. The class of assignments, say Σ^{Const}, for this propositional L-context with constants will thus by definition satisfy the condition that for all $c \in$ ***Const***, $\sigma, \sigma' \in \Sigma^{Const}$

$$\sigma(c) = \sigma'(c).^{19}$$

This, in turn, entails that

$$\boldsymbol{B}\Big|_{\mathcal{S}^{Const}_{\equiv}}\{\,\{\neg\}, \{\wedge, \vee, \rightarrow\}, \boldsymbol{Var}, \boldsymbol{Const_T}, \boldsymbol{Const_F}\} \text{ – where}$$

$$\boldsymbol{Const_T} = \{c_i \in \boldsymbol{Const}: \sigma(c_i) = T\}$$
$$\boldsymbol{Const_F} = \{c_i \in \boldsymbol{Const}: \sigma(c_i) = F\}$$

– which means that *semantic* equifunctionality, even though it is unable do differentiate logical connectives any further than its syntactic precursor, does manage to classify *constants* according to what they denote.

3.2. Semantic Valuation Rules

Propositional L-context involve, apart from the system $\mathcal{S}$ of semantic formation rules (concerned with the proper assignment of denotations to propositional variables) a further kind of semantic rules which I shall refer to as *valuation rules.*

The system of classical valuation rules of $\boldsymbol{L}_0$, say $\mathcal{V}$, determines a function $\bar{v}$: ***Scope***$(\mathcal{S}) \rightarrow$ **Val** which is most economically defined by making use of the characterisation of semantically well-formed formulae in terms of truth-value assignments, i.e. by recursively defining a function v: **Scope** $(\mathcal{G}) \times \Sigma \rightarrow$ **Val** – which has one and the same value for all pairs characterising the same $\mathcal{S}$-formula[20] – and stipulating that $\bar{v}$ $(k\langle\Phi, \sigma\rangle) = v(\Phi, \sigma)$. The recursive definition in question is, of course, that (for $\Phi, \Psi, \Psi' \in$ **Scope** $(\mathcal{G})$, $p \in$ ***Var*** and $\sigma \in \Sigma$)

i) if $\Phi = \langle p\rangle$ then $v(\Phi, \sigma) = \sigma(p)$

ii) if $\Phi = \neg\Psi$ then $v(\Phi, \sigma) = \begin{cases} T \text{ if } v(\Psi, \sigma) = F \\ F \text{ else} \end{cases}$

[19] Note that in the case of propositional L-contexts with constants, assignment functions range over ***Var*** $\cup$ ***Const.***

[20] $k\langle\Phi, \sigma\rangle = k\langle\Phi', \sigma'\rangle \Rightarrow v(\Phi, \sigma) = v(\Phi', \sigma')$

$$\text{iii) if } \Phi = \Psi \wedge \Psi' \text{ then } v(\Phi, \sigma) = \begin{cases} T \text{ if } v(\Psi, \sigma) = T \text{ and } v(\Psi', \sigma) = T \\ F \text{ else} \end{cases}$$

$$\text{iv) if } \Phi = \Psi \vee \Psi' \text{ then } v(\Phi, \sigma) = \begin{cases} T \text{ if } v(\Psi, \sigma) = T \text{ or } v(\Psi', \sigma) = T \\ F \text{ else} \end{cases}$$

$$\text{v) if } \Phi = \Psi \rightarrow \Psi' \text{ then } v(\Phi, \sigma) = \begin{cases} F \text{ if } v(\Psi, \sigma) = T \text{ and } v(\Psi', \sigma) = F \\ T \text{ else} \end{cases}$$

The product of $\mathcal{V}$,[21] i.e. the valuation function $\bar{v}$ is, like any function, set-theoretically representable as a set of ordered pairs, namely $ext(\bar{v}) \subset$ **Scope** $(\mathcal{S}) \times$ **Val**, and $\mathcal{V}$ can accordingly be interpreted as a rule system for transforming $\mathcal{S}$-formulae into a new kind of semantic individuals, which might be called "(classically) *valuated semantically well-formed formulae*", and which, collectively, make up the scope of $\mathcal{V}$, i.e. we can assume that **Scope** $(\mathcal{V}) = ext(\bar{v})$. Given the structure of these individuals – i.e. their being ordered pairs of $\mathcal{S}$-formulae and truth-values – the natural conception of an exchange of basic L_0-expression in such a pair $\langle \Theta, v \rangle \in$ **Scope** $(\mathcal{V})$ must clearly be that

$$\langle \Theta, v \rangle \|^a_b = \langle \Theta \|^a_b, v \rangle.$$

This can easily be extended to all elements of $\mathbf{F}_2 := \mathbf{F}_1 \times$ **Val**, and it will not be surprising that this super-set $\mathbf{F}_2$ of **Scope** $(\mathcal{V})$ is an Ajdukiewicz-field for $\mathcal{V}$ relative to $\boldsymbol{B}$. The "*logical*" equifunctionality relation $\overset{\mathcal{V}}{\equiv}$ – given by

$$a \overset{\mathcal{V}}{\equiv} b \text{ iff}_{\text{df}} \textbf{Scope}(\mathcal{V})\|^a_b = \textbf{Scope}(\mathcal{V})$$

– then imposes the following "logical equifunctionality structure" on our vocabulary:

$$\boldsymbol{B}|_{\overset{\mathcal{V}}{\equiv}} = \{\{\neg\}, \{\wedge\}, \{\vee\}, \{\rightarrow\}, \boldsymbol{Var}\}.$$

This means that $\overset{\mathcal{V}}{\equiv}$ manages to differentiate the logical connectives in precisely the same way as the intuitive relation of synonymy. And indeed, there are strong reasons for adopting $\overset{\mathcal{V}}{\equiv}$ as the set-theoretic correlate to synonymy

[21] Note, incidentally, that all the logical connectives of L_0 are truth-functional relative to $\mathcal{V}$, by which I mean that for any logical connective a there is a function τ_a : $\mathbf{Val}^n \rightarrow \mathbf{Val}$ (where n = rank of a) such that (for any $\Phi_1, \ldots, \Phi_n \in$ **Scope** $(\mathcal{S})$, $\sigma \in \Sigma$)

$$\tau_a(v(\Phi_1, \sigma), \ldots, v(\Phi_n, \sigma)) = v(a(\Phi_1, \ldots, \Phi_n), \sigma).$$

Indeed usually there will be at most one such τ_a for any logical connective a. In our classical framework τ_a is called the *truth table* of a.

(at least for propositional L-contexts with *truth-functional* connectives) – assuming the traditional view that the meaning of classical logical connectives is given in their truth-tables – for it can be shown that any truth-functional logical connectives a and b of a propositional L-context are logically equifunctional (have the same "logical use") relative to the relevant valuation rules if and only it they have the same truth-table; i.e. that

$$a \overset{\mathcal{V}}{\equiv} b \text{ iff } \tau_a = \tau_b.$$

Moreover, the fact that one can show for certain syntactically well-formed formulae, such as '$p_1 \rightarrow p_2$', '$\neg p_1 \vee p_2$' and '$p_1 \vee p_2$', [22] that their logical equifunctionality – namely

$$p_1 \rightarrow p_2 \overset{\mathcal{V}}{\equiv} \neg p_1 \vee p_2 \overset{\mathcal{V}}{\not\equiv} p_1 \vee p_2$$

– corresponds to what I consider to be their intuitive synonymy, gives strong support to the view that our logical equifunctionality relation provides a general explication for the synonymy in $\boldsymbol{L}_0$.

To sum up: given that our analysis in terms of $\boldsymbol{L}_0$ can easily be extended to cover propositional L-contexts in general, we have thus shown that the three hierarchically ordered (formation-) rule systems $\mathcal{G}$, $\mathcal{S}$ and $\mathcal{V}$ involved in propositional L-contexts can set-theoretically be represented in a way which not only i) allows us to define three analogously ordered representations of the relevant sameness-of-use relations, but also ii) gives us reason to adopt the "highest-ranking" of these equifunctionality-relations as set-theoretical explication of the synonymy between the linguistic expressions of propositional L-contexts. Can these results be generalised beyond the propositional case? The answer is, I believe, yes – at least as far as (first-order) predicate L-contexts are concerned. However, this will have to remain a conjecture to be dealt with in a later essay.

To conclude this discussion of semantic rules, let us briefly consider how Ajdukiewicz's own account of synonymy, given in terms of language matrices (see 1.2.1.), might fit in with our results. To do this, let me introduce the notion

[22] The examples chosen here allow for a characterisation of their exchange products without having to introduce an auxiliary system, namely

$$k\langle \Phi_1, \sigma\rangle \|{}^{\Phi}_{\Psi} = k\langle \Phi_1 \|{}^{\Phi}_{\Psi}, \sigma\rangle.$$

A discussion of the logical equifunctionality of wffs in general is, however, technically more demanding and cannot be given in this introductory essay.

of an *Ajdukiewicz-matrix* (based on a vocabulary $\boldsymbol{B}$ and a syntactic rule system $\mathcal{S}$) by which I mean a sub-set of

$$\mathbf{F}_3 := \{\langle\{\Phi_1, \ldots \Phi_{k-1}\}, \Phi_k\rangle : \Phi_i \in \mathbf{F}_0\}^{23}$$

– say, for example,

$$\mathbf{M} := \{\langle\{\Phi_1, \Phi_2\}, \Phi_3\rangle, \langle\emptyset\ , \Phi_4\rangle\} –$$

whose elements contain only syntactically well-formed formulae – which, in the case of **M** means that $\Phi_1, \ldots, \Phi_4 \in$ **Scope** $(\mathcal{S})$.

a) Now, **M** can be interpreted as itself being the scope of a rule system, say $\mathcal{D}$, which has $\mathbf{F}_3$ as its constructive field. This, I believe, is precisely what Ajdukiewicz had in mind with regard to his discursive language matrices, where $\mathcal{D}$ is a system of rules for the construction of *derivations.* $\mathbf{F}_3$ is an Ajdukiewicz-field for $\mathcal{D}$ relative to $\boldsymbol{B}$ and the equifunctionality relation $\overset{\mathcal{D}}{\equiv}$ reflects the sameness of use as far as the constructions (i.e. the derivations) guided by $\mathcal{D}$ are concerned.

b) A second way of interpreting **M** is to treat it as the "linguistic expression" of a system $\mathcal{M}$ of *higher order* semantic rules – by which I mean a system of rules for constructing semantically well-formed formulae of a certain type, which collectively make up the scope of $\mathcal{M}$. That is, we can treat **M** not as the scope of a rule system, but as the "linguistic correlate" to (the scope of) a higher-order system, namely to (the scope of) $\mathcal{M}$.[24]

Which of these interpretations ought to be adopted in order to explicate the relevant notion of synonymy? In my view it must be the second one. Whether sameness of meaning can indeed be explicated in terms of the equifunctionality relation associated with $\mathcal{M}$ is a question which will have to be

[23] Ajdukiewicz matrices correspond to the sort of matrices Ajdukiewicz would associate with what he calls "purely discursive languages", i.e. languages without "empirical meaning rules" in his sense. His distinction between axiomatic and deductive meaning-rules is reflected in whether the set off wffs in a given element of $\mathbf{F}_3$ is empty or not.

[24] Each element of **M** can, for example, be associated with a particular class of models (to use the terminology of predicate L-contexts) – namely

$$\langle\{\Phi_1, \Phi_2\}, \Phi_3\rangle \longleftrightarrow \{\mathsf{M} : \text{if } \mathsf{M} \models \Phi_1, \Phi_2 \text{ then } \mathsf{M} \models \Phi_3\}$$
$$\langle\emptyset, \Phi_4\rangle \longleftrightarrow \{\mathsf{M} : \mathsf{M} \models \Phi_4\}$$

– and the correlation between **M** and $\mathcal{M}$ might be given in the fact that **Scope**$(\mathcal{M})$ is precisely the class of semantically well-formed formulae which involve only assignements of denotations occurring in each and everyone of these classes of models associated with the elements of **M**.

postponed to a future analysis of, say, predicate L-contexts. What seems clear from our discussion so-far (say of the equifunctionality of propositional constants) is that the suggested conception of meaning rules as higher-order semantical rules will be immune against the sort of objection raised by Tarski, and that it will do more justice to the connections between semantic and syntactic features involved in synonymy than the remedy put forward by Giedymin and Ernest.[25]

4. Carnap's Intensional Isomorphism

In the first chapter of *Meaning and Necessity*, Rudolf Carnap develops his *method of extension and intension* "for the semantical analysis of meaning, that is [. . .] for analysing and describing the meaning of linguistic expressions."[26] The importance of Carnap's work and its similarity in overall aim to the work presented here warrants a concluding comparison of the two enterprises, at least as far as such a comparison is possible given the restricted scope of this introductory discussion of the *method of equifunctionality*.

4.1. Semantical Language Systems

A Carnapian *semantical language system* is given by "laying down the following kinds of rules: (1) rules of formation, determining the admitted forms of sentences; (2) rules of designation for the descriptive constants; (3) rules of truth [. . .]; (4) rules of ranges [. . .]."[27] By *rules of formation* Carnap clearly means the sort of rules we called syntactic formation rules, but what are the other types he mentions?

The examples of *rules for designation* which he provides for the particular system S_1 used in his discussion[28] are stipulations in which he assigns the meaning of an ordinary English expression to the non-logical constants of the vocabulary in question, by using a pre-theoretical notion of translation. Thus we are given the rules that

(RD_1) 's' is a symbolic translation of 'Walter Scott', and
(RD_2) 'B*x*' is a symbolic translation of '*x* is a biped'.

[25] See 1.2.2.
[26] Carnap [1956], p. iii.
[27] p. 5.
[28] S_1 is a system based on the first-order predicate grammar with lambda and iota-operators, individual constants and (finitely many) predicate constants.

Based on these designations, the *rules of truth* specify what it is for a sentence to be true in the semantical language system in question. Carnap gives the following examples:[29]

(RT_1) The sentence 'Bs' is true if and only if Scott is a biped.

(RT_2) A sentence $\Phi_i \vee \Phi_j$ is true in S_1 if and only if at least one of the two components is true.

(RT_3) A sentence $\Phi_i \leftrightarrow \Phi_j$ is true in S_1 if and only if either both components are true or both are not true.

Finally, the *rules of ranges*. "A class of sentences in [a semantical language system *S*] which contains for every atomic sentence either this sentence or its negation, but not both, and no other sentences is called a ***state-description*** in [*S*]. [...] state descriptions represent Leibniz's possible worlds or Wittgenstein's possible states of affairs."[30] Given this, the rules of ranges are taken to be the semantical rules which determine whether or not a given sentence ***holds in*** a given state-description. "That a sentence holds in a state-description means, in nontechnical terms, that it would be true if the state-description (that is, all sentences belonging to it) were true. A few examples will suffice to show the nature of these rules:

$[(RR_1)]$ an atomic sentence holds in a given state-description if and only if it belongs to it;

$[(RR_2)]$ $\neg\Phi_i$ holds in a given state-description if and only if Φ_i does not hold in it;

$[(RR_3)]$ $\Phi_i \vee \Phi_j$ holds in a state-description if and only if Φ_i holds in it or Φ_j or both;

$[(RR_4)]$ $\Phi_i \leftrightarrow \Phi_j$ holds in a state-description if and only if either both Φ_i and Φ_j or neither of them hold in it; [...]."[31]

The class of state-descriptions in which a given sentence holds is called the *range* of that sentence (in the language system in question). By determining the ranges, the rules of ranges, together with the rules of designation are taken by Carnap to give "an *interpretation* for all sentences in [the system in question], since to know the meaning of a sentence is to know in which of the possible cases it would be true and in which not [...]"[32] That is to say,

[29] See p. 5.
[30] p. 9.
[31] ibid.
[32] pp. 9 f.

(1) "the meaning of a sentence, its interpretation, is determined by the semantical rules (the rules of designation and the rules of ranges [. . .])"[33]

Given these informal characterisations of the meaning of sentences in a given semantical language system *S*, Carnap then proceeds to explicate the "familiar but vague concept of logical or necessary or analytic truth [. . .]."[34] For this purpose he introduces the technical term of being *L-true*, for which he adopts the following informal general condition – or, as he calls it,

(2) "*Convention.* A sentence Φ_i is ***L-true*** in a semantical system *S* if and only if Φ_i is true in *S* in such a way that its truth can be established on the basis of the semantical rules of the system *S* alone, without any reference to (extra-linguistic) facts."[35]

And he shows that the formal implementation of the Leibnizian conception that a necessary truth must hold in all possible worlds – given in the

(3) "*Definition.* A sentence Φ_i is ***L-true*** (in [a semantical system *S*]) iff$_{\mathrm{df}}$ Φ_i holds in every state-description (in [*S*])."[36]

– is in accord with this convention. From this he proceeds to define the *L*-equivalence for certain sentential components – say in the case of his system S_1 for predicate constants P_i, P_j, on the one hand, and individual constants c_i, c_j, on the other – by using the additional conventions that

(C1) '$P_i \leftrightarrow P_j$' is short for '$(\forall x)(P_i x \leftrightarrow P_j x)$'; and
(C2) '$c_i \leftrightarrow c_j$' is short for '$c_i = c_j$';

in the general definition schema

(D1) *X* is *L*-equivalent to *Y* (in a system *S*) iff$_{\mathrm{df}}$ '$X \leftrightarrow Y$' is *L*-true (in *S*).

This formal explication of *necessity* is then used to define the notion of *having the same intension*:

[33] p. 10.
[34] ibid.
[35] ibid. Note here the link between the meaning of a sentence and a sentence being L-true which Carnap establishes by refering in both cases to the characteristic of being determined by the semantical rules alone.
[36] p. 10.

(D2) X has *the same intension* as (is *co-intensional* with) Y (in a system S) iff$_{df}$ X is *L*-equivalent to Y (in S).

Carnap realises that "for the explication of certain customary concepts [such as 'synonymy'] a stronger relation than identity of intension seems to be required. [...] For example, if we ask for an exact translation of a given statement, say the exact translation of a scientific hypothesis [...] we should usually require much more than agreement in the intensions of the sentences, that is, *L*-equivalence of the sentences. [...] it will be required that at least some of the component designators be *L*-equivalent, in other words, that the intensional structures be alike or at least similar."[37] Accordingly, he introduces the relation of *intensional isomorphism* between sentences, i.e. an isomorphism with respect to their forms (grammatical structure) and the intensions of the corresponding sentential components, and he suggests that "synonymy [...] is explicated by intensional isomorphism."[38] At this point, it is worth noting for our comparative purposes that:

i) Carnap's explication of *synonymy* is "logic-based" in the sense of being based on a formal explication of the logical notion of *necessity* given by his conception of *L*-equivalence; and

ii) to apply Carnap's notion of sentences having the same intension, all we need are the *rules of formation* and the *rules of ranges* of the system in question, while an application of his logically explicated notion of synonymy requires, in addition, the relevant conventions of the sort exemplified in (C1) and (C2).

4.2. *The Sentential Language System S_0 and the Sentential L-context L_0^**

Let us now consider a simplified version, say S_0, of Carnap's language system S_I and a slightly modified version, say L_0^*, of our propositional L-context L_0 which are sufficiently similar to one another for a cross-identification of the relevant rule systems. What I have in mind are a system and a context both based on the vocabulary with the logical connectives $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$, the (descriptive / denoting) constants c_1, c_2, c_3, ... and with *rules of formation / syntactic formation rules* given by the appropriate modification, say $\mathcal{G}^*$, of our rule system $\mathcal{G}$. The sentences (in Carnap's sense) of S_0 are thus nothing but the well-formed formulae of L_0^* (say **Scope** ($\mathcal{G}^*$)); they are *atomic* iff they are of

[37] pp. 59 f.
[38] p. 64.
[39] p. 57.

the form $\langle c_i \rangle$. To do justice to Carnap's views on how this sort of formulae is to be interpreted, S_0 must be a language system used to talk about some given class of sentences (say **Sent**) which, as such, constitutes the domain of discourse of the system. This "sentential interpretation" of the descriptive symbols of S_0 is clearly in conflict with the truth-value interpretation adopted in our propositional L-context $\boldsymbol{L}_0$. Hence the need for a "sentential re-interpretation" of this context, i.e. the need to re-define the assignements σ, σ', σ'' in $\boldsymbol{L}_0$ as functions from the denoting symbols, not into the set of truth-values, but into this set **Sent** of sentences. This re-definition, in turn, requires a re-interpretation (say $\mathcal{V}^*$) of the $\boldsymbol{L}_0$-system of valuation-rules $\mathcal{V}$, as a rule system specifying a *validity-function*

$$\mathcal{V}^* : \textbf{Scope}\,(\mathcal{G}^*) \rightarrow \{0,1\}$$

on the basis of the class Σ^* of functions

$$\sigma^* : \{\text{atomic } \mathcal{G}^*\text{-wff}\} \rightarrow \{0,1\}.^{40}$$

This allows us to identify within $\boldsymbol{L}_0^*$ the components of S_0 required for an application of the Carnapian notion of sentences having the same intension, namely:

a) the formation rules of S_0 are, by stipulation, identical with the syntactic formation rules of $\boldsymbol{L}_0^*$;
b) the elements of Σ^* can be identified with the state-descriptions of S_0 in the way in which sets can be identified with set-theoretic *characteristic functions*;
c) given this, Carnap's rules of ranges for propositional constructs, i.e. RR_1 to RR_4, are nothing but the rules of $\mathcal{V}^*$.

If we now turn to compare the two explications of synonymy (i.e. to compare the $\boldsymbol{L}_0^*$-equifunctionality-relation $\overset{\mathcal{V}^*}{\equiv}$ with the Carnapian relation of intensional isomorphism for S_0) we will immediately notice that our logical equifunctionality relation $\overset{\mathcal{V}^*}{\equiv}$ differs from the relevant Carnapian relation of intensional isomorphism: while

[40] By adopting the sentential interpretation, $\mathcal{V}$ ceases to be representation of a system of *valuation*-rules.

$$`\neg p_1 \vee p_2' \overset{V^*}{\equiv} `p_1 \rightarrow p_2',$$

'$\neg p_1 \vee p_2$' is not intensionally isomorphic with '$p_1 \rightarrow p_2$',[41] which, I believe, is a point which speaks in favour of the equifunctionality account.

The problems with Carnap's attempt to explicate synonymy, however, are not exhausted by this. In his discussion of intensional isomorphism, Carnap states that "[f]or [. . .] the comparison of intensional structures, it seems advisable to [. . .] take as designators all those expressions which serve as sentences, predicators, functors or individual expressions of any sort."[42] Accordingly he must be able to compare logical constants, i.e. he needs some convention of the sort given in (C1) and (C2) for expressions such as '$\wedge \leftrightarrow \vee$', and '$\wedge \leftrightarrow \neg$'. Intuitively, Carnap seems to adopt the view that two logical constants are co-intensional iff

(*) "any two full sentences of them with the same argument expressions are *L*-equivalent."[43]

The problem here is that, given the general nature of (C1) and (C2), he is required to formulate this intuition as a statement *in the language under investigation*, something which is clearly far beyond the expressive power of L_0^*. Even though it might be possible to alter the format of Carnap's conventions in a way which avoids this impasse – say by admitting formulations of '$X \leftrightarrow Y$' in terms of languages richer than the one under investigation – this would still not remove what I consider to be a fundamental shortcoming in Carnap's account: the lack of a *general* method of how to arrive at formulations for '$X \leftrightarrow Y$' whatever the type of sentential component which X and Y are referring to may be.[44] The equifunctionality explication of synonymy, in contrast,

[41] a) The modifications imposed on L_0 to obtain L_0^* – apart from the involved enlargement of the vocabulary – do not alter the equifunctionality relations we have attributed to wffs and vocabulary expressions in the context L_0.

b) '$\neg p_1 \vee p_2$' and '$p_1 \rightarrow p_2$' are not of the same grammatical structure, for the one has two connectives while the other has only one.

[42] p. 57.

[43] Ibid.

[44] It might be objected that such a method is indeed given by the general requirement that '$X \leftrightarrow Y$' is a symbolic translation of 'X is co-extensional with Y'.
The problem here is that, even though it might be obvious what the extensions of individual constants and of predicates are meant to be, it is by no means self-evident how we are to extend this concept to other sentential components.

ent[45] to the condition of their "truth-tables" being equal, a condition which we have *derived* to be the case iff they are logically equifunctional. In other words, Carnap's intuition concerning the co-intensionality of logical constants is derivable from the notion of synonymy as explicated by the method of equifunctionality.

REFERENCES

AJDUKIEWICZ, K. 1934 a; "Sprache und Sinn"; *Erkenntnis* IV; pp. 100–38.

AJDUKIEWICZ, K. 1934 b; "Das Weltbild und die Begriffsapparatur"; *Erkenntnis* IV; pp. 259–87.

AJDUKIEWICZ, K. 1935; "Die wissenschaftliche Weltperspektive"; *Erkenntnis* V; pp. 22–30.

BELL, J.L. und M. Machover 1977; *A Course in Mathematical Logic*, Amsterdam: North Holland.

CARNAP, R. 1956; *Meaning and Necessity: A Study in Semantics and Modal Logic*; Chicago: Univ. Press.

ERNEST, P. 1985; *Meaning and Intension in Mathematics*; Ph.D. thesis, London University.

GIEDYMIN, J. 1978; "Radical Conventionalism"; in *The Scientific World Perspective*; J. Giedymin (ed.); pp. XIX-LIII.

LAUENER, H. 1982 "Method in Philosophy and Logic", *Dialectica*, 36, 357-71.

LAUENER, H. 1985 "Methodologische Bemerkungen zur Philosophie des Geistes"; in *Philosophy of Mind, Philosophy of Psychology*, Wien, 197-205.

LAUENER, H. 1986 "Die Sprache der Fiktion", in *Erkenntnis*, 24, 343-62.

LAUENER, H. 1987 "Philosophie als normative Tätigkeit (offener Transzendentalismus versus Naturalismus)", *Dialectica*, 41, 28-38.

LAUENER, H. 1990; "Holism and Naturalized Epistemology Confronted with the Problem of Truth"; in R. Barrett und R. Gibson, eds., *Perspectives on Quine*, Oxford: Blackwell, 213-28.

LAUENER, H. 1990 "Transzendentale Argumente pragmatisch relativiert: Über fundamentale Optionen in der Philosophie", *Erkenntnis*, 33, 223-49.

LAUENER, H. 1992 "Speaking about Language: On the Nature of Linguistic Individuals", in A.P. Martinich and M. White, eds., *Certainty and Surface in Epistemology and Philosophical Method*, New York: E. Mellen Press, 117-34.

LAUENER, H. "Offene Transzendentalphilosophie (Methodologie und pragmatisch relativiertes Apriori)"; to appear in H. Stachowiak, ed., *Pragmatik*, Bd. V, Hamburg; Meiner Verlag.

[45] Provided the constants are "truth-functional" in the sense of 3.2.1. (suitably modified for the sentential interpretation of L^{*}_{0}).