Logique & Analyse **147-148** (1994), 339-366

A MINIMAL LOGICAL SYSTEM FOR COMPUTABLE CONCEPTS AND EFFECTIVE KNOWABILITY*

Max A. FREUND

I. *Introduction*

In contradistinction to logical realism and nominalism, conceptualism, as a philosophical theory of predication, posits concepts as the semantic grounds for the correct or incorrect application of predicate expressions. Then, such a theory assumes that concepts constitute the semantic grounds for predicate expressions to be false or true of things. A modern version of conceptualism, stated and explored for example in *Cocchiarella* (1986a), (1986b), (1989) and (1993), is conceptual realism. This form of conceptualism maintains a dispositional view of concepts, more precisely, it looks at concepts as *cognitive* (human) *capacities,* or cognitive structures otherwise based upon such capacities, to identify, characterize, classify or relate objects. Conceptual realism, we should point out, presupposes an ontological distinction between objects and concepts, which is reflected, according to such a philosophical theory, in their semantic relation to expressions of the language: predicate expressions can never stand for objects, only for concepts; singular terms can never denote concepts, only objects. Concepts have an unsaturated nature which consists in their dispositional status as a cognitive capacity, while objects, on the contrary, possess a saturated nature. [For details on the nature of unsaturateness and saturateness of concepts and objects, respectively, see Cocchiarella (1993)].

An additional feature of conceptual realism is related to the nominalization of predicate expressions, i.e., the transformation of predicate expressions into abstract singular terms, such as the transformation of "human" and "red" into "humanity" and "redness", respectively. Conceptual realism does not assume that every nominalization denotes but only that some of them do it, in which case, according to this philosophical theory, what they denote is an object. However, there is no assumption concerning the nature of such an object, apart from the requirement that there must be a connection between the object denoted by the nominalized predicate expression

* I am grateful to the referee for helpful comments on a previous version of this paper.

and the concept such a predicate expression stands for. The connection should be based on the object being somehow related to the truth conditions determined by the possible applications of the concept. For this reason, possible denotations of nominalized predicate expressions are also called *concept-correlates*.

Cocchiarella has not introduced within conceptual realism a distinction between concepts which are effective capacities and those which are not. By a *concept as an effective capacity* we shall understand a concept whose exercise by any agent will allow him to identify, characterize or relate objects in various ways in a finite amount of time, without the agent resorting to random devices or to his ingenuity. Also, exercise of such a concept always yields the same result, that is, objects declared by an agent to fall under a given concept of that sort will always be declared to fall under the same concept, by the same agent or any other agent. We shall hereafter also refer to concepts which are effective capacities as computable concepts.

Among concepts which are effective capacities or computable, we shall distinguish between (what we shall call) *fully computable concepts and semi-computable concepts*. By *fully computable concepts* we shall mean computable concepts whose exercise will determine whether or not given objects fall under them. More clearly, exercise of a fully computable concept will allow the agent to determine in a finite amount of time, without resorting to random devices or to his ingenuity, whether or not given objects should be identified, characterized or related in the ways the concept does it. The concept of a number being the addition of two numbers and the concept of a number being the product of two numbers are clear examples of (fully) computable concepts. By *semi-computable concepts* we shall mean computable concepts whose exercise will determine that given objects fall under them whenever they do actually fall. In other words, exercise of a semi-computable concept will allow the agent to determine in a finite amount of time, without the agent resorting to random devices or to his ingenuity, whether given objects should be identified, characterized or related in the ways the concept does it. Consequently, if a predicate expression stands for a (fully or semi) computable concept, then such a concept will provide an effective rule for the application and/or non-application of the predicate expression.

Our concern, in this paper, is with the logical properties of computable concepts as well as with the logical features of knowledge obtainable through such concepts, which we shall call *effective knowledge*. To this end, we shall characterize a formal language whose logical syntax contains as primitive symbols, among others, a propositional operator and a quantifier formally representing, respectively, the epistemic operator "it is effectively knowable that p" and a universal quantifier whose range is the set of semi-computable concepts. We shall offer an intuitive interpretation of the epistemic operator and, on its basis, we shall formulate an epistemic second order logical system, which we shall call *MEK*. This system captures important logical principles governing effective knowability and the universal quantifier over semi-computable concepts. Indeed, in terms of this quantifier, we shall define another quantifier formally representing the universal quantifier whose range is the set of fully computable concepts and we shall prove, within *MEK*, logical theses valid for this defined symbol.

We shall also show the relative consistency of $MEK + AC^s$, i.e., the system resulting from the addition of AC^s to the axiomatic base of *MEK*, where AC^s is a given formula expressing the statement that every semi-computable concept has a correlate. This statement will be justified by appealing to both so called *Church's Thesis* and our proposal, in this paper, of interpreting correlates of computable concepts as *Turing machines*. Now, AC^s do not express our view of semi-computable concepts correlates as Turing machines. So, in order to express this interpretation, we shall consider an applied form of *MEK* to the language of arithmetic as well as add a set *Q* of axioms to the axiomatic base of *MEK*. We shall call this system *Q-MEK* and show that within *Q-MEK* it is possible to construct a formula (which we shall call T^s) consistent with *Q-MEK* and, obviously with *MEK* as well, in which the above interpretation of computable concepts correlates is explicitly stated. Finally, the relative consistency of some of the most philosophically important extensions of $MEK + AC^s$ as well as of those of $Q\text{-}MEK + T^s$ will be also shown.

As a final note, we should point out that in Shapiro(1985), Reinhardt (1985) and Reinhardt (1986) projects of constructing formal systems of arithmetic and/or set theory containing a propositional epistemic operator, closely related to our operator, have been explored. In the case of Shapiro's (cf. pp. 6, 11), the epistemic operator is intuitively interpreted as "it is ideally or potentially knowable that *p*" where this notion is to be understood as meaning "knowledge obtainable (in principle) by means of constructive methods" cf. *Shapiro* (pp. 6, 11). Although not totally clear in such a work, constructive methods would seem to include, at least, intuitionistic as well as effective methods. Now, even though such an epistemic operator would clearly bear a certain relationship to our operator (which will be formally represented in our logical system by "$[e]$"), the intuitive interpretation of this operator is different from Shapiro's. On one hand, our interpretation pressuposes a philosophical framework which is not assumed in such a work. On the other hand, the logic of the operator in Shapiro is S4, whereas, as it would emerge from our discussion in the following sections of this article, $[e]\varphi \rightarrow [e][e]\varphi$ could be not a plausible principle. According to our discussion in sections II and III, effectively knowing that φ implies exercise of computable concepts as a necessary and sufficient condition in justifying our belief in φ. Now, effectively knowing that φ is effectively knowable will necessarily require *effectively* knowing that the concepts involved in the effective justification of φ are in fact computable. However, in view of a well known result according to which the set of algorithms is not effectively enumerable [*see Davis* (1958, pp. xvi-xviii, 77-8)] and our view in section II concerning the link between computable concepts and algorithms , effectively knowing that a given computable concept is computable might never be possible. Therefore, even if φ were effectively knowable, it might not be the case that it is effectively knowable that φ is effectively knowable. Although φ might be effectively knowable, resorting to evidence that requires exercising computable concepts as sufficient and necessary condition might never be sufficient to justify the belief that φ is effectively knowable. This is because it might never be possible to effectively know that the concepts involved in an effective justification of φ are computable. Similar remarks on the difference between our project and Shapiro's equally applies to Reinhardt's.

II. *Algorithms, computable concepts, Turing machines and concept-correlates*

As usual, we shall distinguish an *intuitive* sense from *formal* senses of computability such as *Turing-computability*, *recursivity* and *lambda definability*. It is well known that all formal senses so far proposed have been proved to be extensionally equivalent. [For details, see, for example, *Boolos-Jeffrey*(1980), *Davis*(1958) or *Kleene*(1952)]. A notion essential to intuitive computability is that one of an an *algorithm*, that is, a *finite set* of *mechanical deterministic instructions which can give an answer in a finite amount of time* given certain input-data. A problem or set of problems with respect to a certain entity is characterized as intuitive computable if and only if there is an algorithm for solving that problem or set of problems with respect to that entity. The problem of the values of a function given certain arguments, for example, is said to be intuitively computable if and only if there is an algorithm which, for any given argument or arguments, the algorithm can calculate the value of the function (if there is such a value) for such an argument or arguments.

Now, concerning algorithms and computable concepts, we shall look at them as essentially linked to each other, in the sense that there necessarily is an algorithm corresponding to a computable concept and vice versa. This is because, on one hand, computable concepts provide the cognitive basis for apprehending an algorithm as a procedure for computing a problem or set of problems with respect to a certain entity, such as a numerical function. It is well known that the values of the same numerical function can be computed by different algorithms. The values of addition, for example, can be shown to be computed by denumerably many different Turing-machine algorithms, Lambda-calculus algorithms or Unlimited registers-machine algorithms; but all of these possible algorithms are associated to the same function because they are cognitively apprehended as computing the values of that function. Without such an apprehension the different algorithms are just sets of instructions which, given the same input-data, yield the same output-data, that is, no link is established between the algorithms and the function. The apprehension is possible only if a computable concept of the numerical function has been constructed. If no concept of a number being the addition of another two numbers has been formed, it is no possible to apprehend an algorithm as computing the values of addition.

On the other hand, any computable concept can be associated, in principle, with an algorithm. More clearly, when the content of a computable concept is grasped, then, *in principle*, it is possible to formulate an algorithm whose implementation will classify, identify or relate objects in the same way as it is done by exercising the concept associated with the algorithm. The algorithm will express in its set of instructions the rule provided by the content of the concept. For example, once the content of the concept of a number being the addition of another two numbers is grasped, it is possible, in principle, to formulate an algorithm yielding the addition of two numbers. Also, different algorithms can be associated with the same concept, that is, computable concepts can, as it were, *modeled* by different effective procedures.

The connection between the intuitive and the formal senses of computability has been established in the so called *Church's Thesis*. According to this thesis *intuitive computability* is extensionally equivalent to *Turing-computability* (or to any other formal sense of computability so far proposed). Consequently, if an algorithm or effective procedure for solving a problem or set of problems with respect to a certain entity exists, then a Turing-algorithm must exist for the same problem or set of problems and entity and vice-versa. Assuming Church's thesis as well as our view concerning the link between algorithms and computable concepts, it follows that for any Turing-computable problem with respect to a certain entity, computable concept or concepts can be formed whose exercise would make possible human computation of the same problem. Also, if computable concept or concepts can be formed which make human computation of a given problem with respect to a certain entity possible, then there should be a Turing machine (*T-machine*, for short) whose implementation would compute the problem.

On the basis of Church's Thesis and our above line of thought, we shall adopt T-machines as correlates of computable concepts. Clearly, a T-machine is an abstract object, namely: a *set* of quadruples (satisfying certain conditions) from a given countable alphabet. It is also related to the concept truth-conditions by expressing in its set of quadruples the effective rule provided by the concept. Then, our interpretation of concept correlates in terms of T-machines agrees with the general approach, in conceptual realism, of viewing such entities as objects of certain sort, somehow related to the truth conditions determined by the possible application of the concept. Since it is possible to have denumerably many equivalent T-machines (that is, machines yielding the same output-data to the same input-data), then in such cases it should be chosen as a concept-correlate the T-machine having the smallest gödel number under a suitable arithmetization of the theory of T-machines such as that one in *Davis* (1958, pp. 56-7). It is important to notice that correlates for non-computable concepts will not be postulated, since algorithms cannot be associated with such concepts in the sense pointed out above with respect to computable concepts.

III. *Effective Knowability*

We proceed now to describe several features we shall consider to be intuitively involved in the concept of effectively knowability, that is, in the concept of that which can, *in principle*, be effectively known. We shall approach knowledge from a classical perspective and assume its main thesis, according to which a *necessary* condition for possessing knowledge (of a certain statement p) is to have a true justified belief in p. Consequently, we shall assume that a necessary condition for *effectively* knowing a certain statement p is to have an *effectively* justified true belief in p. By an *effectively justified belief* we shall understand a belief which can be justified by resorting to evidence that requires, as a *necessary* and *sufficient condition*, exercising computable concepts. That is, exercise of only computable concepts allows the knower to obtain evidence that will effectively justify his belief in the truth of p and effectively justifying his belief in φ will necessarily require exercising computable concepts.

Effective justification might pressupose exercise of computable concepts the knowing agent has not formed yet but which can be formed, once certain limitations such as time were put aside. Also, exercise of computable concepts might require an amount of time, scratch paper on which to write the results of progressively exercising computable concepts and a level of technical development the agent might not possess. If these restrictions were put aside, the agent might be able to fully exercise those concepts so as to obtain the evidence which will justify his belief. In all of these cases, we find a situation in which effective knowledge can be achieved once certain limitations of the knower are put aside. It is in terms of this situation that we shall interpret the concept of effective knowability. More precisely, by effective knowledge in principle (or effective knowability) we shall understand effective knowledge the knower can have access to if his restrictions related either to formation of concepts or to exercise of concepts were put aside.

We should note that our assumptions in both *sections* I and II (according to which there is a sense in which computable concepts are to be necessarily associated with effective procedures or algorithms) furnishes us with a ground to approach effective knowledge in terms of algorithms as well: by implementing algorithms associated with computable concepts, evidence can be obtained to justify the agent's belief in a given statement. Now, as in the case of computable concepts, implementation of algorithms might require capacities which go beyond those possesed by the knowing agent (such as large amounts of memory capacity, technical development and capacity to manage complex calculations). Putting aside these limitations, the agent might be able to implement the algorithms.

IV. *Logical Syntax*

We shall now characterize the logical syntax of certain sort of formal languages containing, among its logical symbols, the propositional operator "$[e]$", which is intended to formally represent effective knowability. Such languages will also contain the lambda operator to allow for the formal representation of complex predicate expresions as well as universal quantifiers applicable to individual and predicate variables. Clearly, the syntax of the languages is of a second order nature.

We take a language L to be a countable set of individual and predicate constants. We assume the availability of denumerably many individual variables as well as denumerably many n-place predicate variables (for each natural number n). We shall also use x, y, z and w, with or without numerical subscripts, to refer in the metalanguage to individual variables and F^n, G^n and R^n to refer to n-place predicate variables. We shall usually drop the superscript when the context makes clear the degree of a predicate variable or when otherwise does not matter what degree it is. For convenience, we shall also use u in order to refer to variables in general. As primitive logical constants we take $\rightarrow, =, \sim, \lambda, \forall, \forall^s$ and $[e]$, which we shall intuitively interpret, respectively, as the material implication, identity, classical negation, the lambda abstract operator, the universal quantifier, the universal quantifier for semi-computable concepts and, as noted above , the epistemic operator "it is effectively knowable that p".

Given a language L, we recursively define the set of meaningful expressions of type n of L, (in symbols, $ME_n(L)$ as follows:

(1) every individual variable or constant is in $ME_0(L)$
every n-place predicate variable or constant is in both $ME_{n+1}(L)$ and $ME_0(L)$
(2) if $a, b \in ME_0(L)$, then $(a = b) \in ME_1(L)$
(3) if $\pi \in ME_{n+1}(L)$ and $a_1, \ldots, a_n \in ME_0(L)$, then $\pi(a_1, \ldots, a_n) \in ME_1(L)$
(4) if $\delta \in ME_1(L)$ and $x_1, \ldots, x_n$ are pairwise distinct individual variables, then $[\lambda x_1, \ldots, x_n \delta] \in ME_{n+1}(L)$
(5) if $\delta \in ME_1(L)$, then $\sim \delta \in ME_1(L)$.
(6) if $\delta, \sigma \in ME_1(L)$, then $(\delta \rightarrow \sigma) \in ME_1(L)$.
(7) if $\delta \in ME_1(L)$, then $[\mathrm{e}]\delta \in ME_1(L)$.
(8) if $\delta \in ME_1(L)$, x is an individual variable and F is a predicate variable, then $\forall F\delta, \forall^s F\delta, \forall x\delta \in ME_1(L)$.
(9) if $\delta \in ME_1(L)$, then $[\lambda\delta] \in ME_0(L)$.
(10) if $n > 1$, then $ME_{n+1}(L) \subseteq ME_0(L)$.

We set $ME(L) = \bigcup_{n \in \omega} ME_n(L)$, that is, the set of meaningful expressions of L. We shall use δ, σ, π and α to refer to meaningful expressions of L.

Whenever $t \in ME_0(L)$, we shall say that t is a term of L. We shall use a, t and b, with or without numerical subscripts, to refer to terms in general. On the other hand, for $n \in \omega$ we shall understand $ME_{n+1}(L)$ to be the set of n-place predicate expressions. The *wffs* (well formed formulas) of L are just the members of $ME_1(L)$. The concepts of a bound and free occurrence of a predicate or individual variable are understood as usual. An occurrence of a term b in a wff or term σ is said to be a *bound occurrence* of b *in* σ if some occurrrence of a variable in b is a free occurrence of that variable in b but a bound occurrence of that variable in σ. If a, b are terms and φ is wff or a term, then by $\varphi(a / b)$ we shall mean the wff or term that results by replacing each free occurrence of b in φ by a free occurrence of a, if such a wff or term exists, and otherwise we take $\varphi(a / b)$ to be just φ itself. We shall say that a is free for b in φ, if $\varphi(a / b)$ is not φ unless a is b. Finally, we shall say that a *wff* δ comes from a wff β by rewriting the bound occurrences of a variable u in a meaningful subexpression γ of β by a variable k of the same type as u if and only if there are meaningful expressions σ and α such that, either:

γ is $\forall u\, \sigma$ and α is $\forall k\, \sigma(k / u)$,
γ is $\forall^s u\, \sigma$ and α is $\forall^s k\, \sigma(k / u)$, or
γ is $[\lambda \mathrm{x}_1 \ldots x_n \sigma]$ and α is $[\lambda \mathrm{y}_1 \ldots y_n \sigma(y_1 / x_1 \ldots y_n / x_n)]$, where $y_m = k$ and $\mathrm{x_m} = u$ and $x_j = y_j$, for some m and for every $j \neq m$ such that and $0 < j \leq n$ and $0 < m \leq n$.

and δ is the result of replacing one or more occurrences of γ in β by α.

The universal quantifier "$\forall$" when applied to predicate variables will be intuitively interpreted as the universal quantifier whose range is the set of all concepts (computable or otherwise), i.e., "$\forall F$" should be read as "for every concept F" and when applied to individual variables should be read as "for every individual". On the other hand, "$\forall^s F$" should be read as "for every semi-computable concept F". We define the existential quantifiers as usual:

$$\exists F\varphi =_{df} \sim \forall F \sim \varphi \qquad \exists x\varphi =_{df} \sim \forall x \sim \varphi, \qquad \text{and } \exists^s F\varphi =_{df} \sim \forall^s F \sim \varphi.$$

We have not introduced a primitive symbol which, on the intended interpretation becomes a universal quantifier whose range is the set of fully computable concepts, because it can be defined by the primitive symbols already introduced. As the reader will notice, this definition will be in accord with our informal characterization (in section I) of fully and semi-computable concepts. But the reader will also notice that it presuposses the possibility that a concept can be formed from two different concepts, one being the logical complement of the other. However, there is no problem with this assumption, since a view in conceptual realism, to which we shall here subscribe, is that concepts can be formed from other concepts by applying booleans operations.

Definitions:

$$\forall^c F\varphi =_{df} \forall F\Big(\big(\exists^s G\big(G = [\lambda x_1 \ldots x_n F x_1 \ldots x_n]\big)$$
$$\& \, \exists^s G(G = [\lambda x_1 \ldots x_n \sim F x_1 \ldots x_n])\big) \to \varphi\Big)$$
$$\exists^c F\varphi =_{df} \sim \forall^c F \sim \varphi$$

In other words, according to this definition, F is a fully computable concept if and only if there are semi-computable concepts G and G^* such that exercise of G will determine that given objects fall under F whenever they do actually fall under F and exercise of G^* will determine that given objects do not fall under F whenever they do not actually fall under F.

V. *A second order epistemic logical system*

We proceed now to formulate a minimal logical formal system for computable concepts and effective knowability as well as sketch proofs for several of its most important theorems. However, we shall first make some brief remarks on the principles (involving the epistemic operator or quantifiers whose range are computable concepts) assumed as axioms by the formal system. We begin by considering principles $[e]\varphi \rightarrow \varphi$ and $[e](\varphi \rightarrow \gamma) \rightarrow [e]\varphi \rightarrow [e]\gamma$. The former principle states the so called "truth-condition" for effective knowability: we have effective knowledge of truth statements only. We explicitly assumed this property in our characterization of effective knowability, in section III. In order to see that the latter principle is valid, just recall that once we effectively determine the truth of a material implication and its antecedent, then there is an effective procedure for deciding the truth of the consequent, that is, through truth table methods.

Another principles we shall adopt as axioms are $[e]\forall x\varphi \rightarrow \forall x[e]\varphi$, $[e]\forall^s F\varphi \rightarrow \forall^s F[e]\varphi$ and $[e]\forall F\varphi \rightarrow \forall F[e]\varphi$. Clearly, if it is effectively knowable that for every entity *e*, φ is the case, then by universal instantiation it is effectively knowable of any particular entity *e* that φ is the case. We should note that the converse of each one of those three principles (*viz.* $\forall x[e]\varphi \rightarrow [e]\forall x\varphi$ $\forall^s F[e]\varphi \rightarrow [e]\forall^s F\varphi$, $\forall F[e]\varphi \rightarrow [e]\forall F\varphi$ might not be intuitively valid in an infinite domain of quantification. That is, for every entity *e* belonging to the domain *D* of quantification (where *D* might be either the set of all possible concepts, the set of all possible semi-computable concepts or the set of individual objects), it can be effectively knowable of *e* that φ, but this by itself, being *D* infinite, only guarantees that at any given moment of time *t* the knowing agent will be able to effectively know of all the entities of the domain *D* considered at or before *t* (but not necessarily of all entities of *D*) that φ.

The role of identity in intensional contexts has been extensively analysed in the literature. It is well known that Leibniz's law cannot hold in its unrestricted form in every intensional context, that is, equals cannot be validly substituted, in general, for each other within intensional contexts, in particular, epistemic contexts. For example, the unrestricted form of that law, in the philosophical framework of this article, would allow us to infer (using propositional logic only and the evident formula $[e](a = a)$), the formula $a = b \rightarrow [e]a = b$. However, this formula cannot be valid, since it will turn any empirical identity into an statement for which an effective justification could (in principle) be obtained. Instead of Leibniz's law, we shall assume the following two formulas: $a = b \rightarrow \varphi \leftrightarrow \psi$ (where φ does not contain the $[e]$ operator) and $[e]a = b \rightarrow (\varphi \leftrightarrow \psi)$, where ψ comes from φ by replacing any free occurrence of *a* by a free occurrence of *b*. In other words, we shall commit ourselves to Leibniz's law with respect to a given identity $a = b$ only when such an identity is effectively knowable, otherwise to a restricted form of such a law to non-intensional contexts.

After our brief remarks above, we shall now state the *minimal logical system for effective knowability* (*MEK*, for short).

System *MEK*

Where φ, ψ, σ, γ are wffs and u a predicate or individual variable, the axioms of *MEK* are

A0. all tautologies

A1. $[e](a=a)$

A2. $\forall u(\varphi \to \psi) \to (\forall u\varphi \to \forall u\psi)$

A3. $\forall^s F(\varphi \to \psi) \to (\forall^s F\varphi \to \forall^s F\psi)$

A4. $\varphi \to \forall u\varphi$, provided u does not occur free in φ.

A5. $[e](a=b) \to (\varphi \leftrightarrow \psi)$, (where ψ comes from φ by replacing one or more free occurrences of a by free occurrences of b).

A6. $(a=b) \to (\varphi \leftrightarrow \psi)$ (where ψ comes from φ by replacing one or more free occurrences of a by free occurrences of b and there is no occurrence of the epistemic operator in φ.[1]

A7. $\forall F^s \exists G^s (F=G)$

A8. $\forall F \exists G(F=G)$

A9. $\forall x \exists y(x=y)$

A10. $\forall F\varphi \to \forall^s F\varphi$

A11. $([\lambda x_1 \ldots \lambda x_n Rx_j \ldots x_n] = R)$

A12. $[\lambda x_1 \ldots \lambda x_n \varphi](a_1 \ldots a_n) \leftrightarrow \exists x_1 \ldots \exists x_n (a_1 = x_1 \& \ldots a_n = x_n \& \varphi)$ (where no x_j is free in any a_k, for $1 \le k,\ j \le n$)

A13. $\sigma \leftrightarrow \sigma^*$, where σ^* comes from σ by rewriting the bound occurrences of a variable u in a subexpression β of σ by a variable of the same type as u new to β.

A14. $[e]\varphi \to \varphi$

A15. $[e](\varphi \to \gamma) \to ([e]\varphi \to [e]\gamma)$

A16. $[e]\forall x\varphi \to \forall x[e]\varphi$

A17. $[e]\forall^s F\varphi \to \forall^s F[e]\varphi$

A18. $[e]\forall F\varphi \to \forall F[e]\varphi$.

Where φ, ψ are *wffs* and u either a predicate or individual variable, the rules of *MEK* are: Modus Ponens (MP): from φ, $\varphi \to \psi$ infer ψ; and Universal Generalization (UG): from φ infer $\forall u\varphi$.

If there is a finite sequence of well formed formulas such that every member of the sequence is either an axiom of *MEK* or follows from previous members of the sequence by one of the rules of *MEK*, then we shall say that the last formula φ of the sequence is a *theorem* of *MEK*, (in symbols $\vdash \varphi$). From now on, every proof of a theorem or derived rule, requiring reasoning in accordance with principles and rules of classical propositional

[1] As the reader might have noted, the proviso that ψ be a wff precludes expressions such as $F=c \to (F(F) \leftrightarrow c(c))$ and $[e](F=c) \to (F(F) \leftrightarrow c(c))$ (where F is a monadic predicate expression and c an individual constant) from being instances of A5 and A6, respectively, since "$c(c)$" would not be a wff.

logic, will be indicated by the expression *PL*. We now proceed to prove several theorems and derived rules.

By *UG*, *PL* and A10, the following rule of generalization for the quantifier over semi-computable concepts can be derived within *MEK*: (UG^s) if $\vdash \varphi$, then $\vdash \forall^s F\varphi$. Obviously, by A10, A8 and *PL*, we can show that (within *MEK*) every semi-computable concept is a concept:

Th.1 $\quad \vdash \forall^s G \exists F(G = F)$

A version of A4 for semi-computable concepts can be proved by *PL*, A4 and A10:

Th.2 $\quad \vdash \varphi \rightarrow \forall^s F\varphi$, provided F does not occur free in φ

Now, concerning universal instantiation principles, only their restricted forms can be proved to be theorems of the system. By *UG*, A5 (or A6 in the case of Th. 4), A2 and *PL*, we can show

Th.3 $\quad \vdash \exists F^n[e](F = [\lambda x_1 \ldots x_n \sigma]) \rightarrow (\forall F^n \varphi \rightarrow \varphi[\lambda x_1 \ldots x_n \sigma] / F)$ and

Th.4 $\quad \vdash \exists F^n(F = [\lambda x_1 \ldots x_n \sigma]) \rightarrow (\forall F^n \varphi \rightarrow \varphi[\lambda x_1 \ldots x_n \sigma] / F)$

(provided F does not occur free in σ, $[\lambda x_1 \ldots x_n \sigma]$ is free for F in φ and, in the case of Th.4, there is no occurrence of the epistemic operator in φ); and by *UG*, A5 (or A6 in the case of Th. 6), A2 and *PL*:

Th.5 $\quad \vdash \exists x[e](x = a) \rightarrow (\forall x\varphi \rightarrow \varphi a/x)$ and

Th.6 $\quad \vdash \exists x(x = a) \rightarrow (\forall x\varphi \rightarrow \varphi a/x)$

(provided x does not occur free in a, and a is free for x in φ). Finally, by definition, UG^s, *PL*, A5 (in the case of Th.7) or A6 (in the case of Th.8) and Th.2:

Th.7 $\quad \vdash \exists^s F^n[e](F = [\lambda x_1 \ldots x_n \sigma]) \rightarrow (\forall^s F^n \varphi \rightarrow \varphi[\lambda x_1 \ldots x_n \sigma] / F)$

Th.8 $\quad \vdash \exists^s F^n(F = [\lambda x_1 \ldots x_n \sigma]) \rightarrow (\forall^s F^n \varphi \rightarrow \varphi[\lambda x_1 \ldots x_n \sigma] / F)$,

(provided F does not occur free in σ, $[\lambda x_1 \ldots x_n \sigma]$ is free for F in φ and, in the case of Th.8, there is no occurrence of the epistemic operator in φ).

Clearly, formulas $\exists x(x = t)$ (for any term t), $\exists F(F = t)$ and $\exists^s F(F = t)$ (for any term t of the same type as F) cannot be derived within *MEK* by universal instantiation principles, since only restricted forms of such principles can be proved as we have noted above. Moreover, the semantics we shall develop in section VI will provide a concept of validity with respect to which a soundness theorem relative to *MEK* can be proved. Since the aforementioned formulas can also be shown not to be valid in this semantics, then, in view of the soundness theorem, those formulas can not be theorems of *MEK* and, therefore, *MEK* is free of existential pressupositions.

By *PL*, *UG* and definition, the rule of universal generalization for computable concepts can be derived within *MEK*, that is, (UG^c): if $\vdash \varphi$, then $\vdash \forall^c F\varphi$. Versions of A4 and A2 for fully computable concepts can also be proved by *PL*, *UG*, A2, A4 (in the case of Th. 9) and by definition on page 347, *PL*, *UG*, A2 (in the case of Th. 10):

Th.9 $\vdash \varphi \rightarrow \forall^c F\varphi$, provided F does not occur free in φ,

Th.10 $\vdash \forall^c F(\varphi \rightarrow \sigma) \rightarrow (\forall^c F\varphi \rightarrow \forall^c F\sigma)$.

We can show (by *PL*, A11, A6, *UG* and definition on page 347) that according to *MEK* every computable concept is semi-computable:

Th.11 $\vdash \forall^c F \exists^s G(F = G)$

and, by this theorem, *PL*, A10, UG^c, Th. 10 we can prove that every fully computable concept is a concept:

Th.12 $\vdash \forall^c F \exists G(F = G)$.

Restricted principles of universal instantiation for the quantifier over fully computable concepts can also be shown (by A5 and A6, UG^c, *PL*, Th 9 and Th. 10) to be theorems:

Th.13 $\vdash \exists^c F^n[e](F = [\lambda x_1 \ldots x_n \sigma]) \rightarrow (\forall^c F^n \varphi \rightarrow \varphi[\lambda x_1 \ldots x_n \sigma] / F)$

Th.14 $\vdash \exists^c F^n(F = [\lambda x_1 \ldots x_n \sigma]) \rightarrow (\forall^c F^n \varphi \rightarrow \varphi[\lambda x_1 \ldots x_n \sigma] / F)$

(provided F does not occur free in σ, $[\lambda x_1 \ldots x_n \sigma]$ is free for F in φ and, in the case of Th 14, there is no occurrence of the epistemic operator in φ). Also, by definition on page 347, *PL*, Th.4, *UG*, A8, A2, we can prove:

Th.15 $\vdash \forall^c F \exists^c G(F = G)$

Consistency of MEK.

Let φ^* be the result of replacing any occurrence of "$\forall^s$" by "$\forall^1$" (i.e., *Cocchiarella's* (1986b) universal quantifier of the first level of predicative concept formation) and of '$[e]$' by "~~" in a wff φ of *MEK*. Then, φ^* is a wff of second order logical system $HRRC^* + Ext^*_\lambda$ [See Cocchiarella (1986b), for details on this system]. It can easily be shown that φ^* is a theorem of $HRRC^* + Ext^*_\lambda$ if φ is a theorem of *MEK*, that is, if $\vdash \varphi$ then φ^* is a theorem of $HRRC^* + Ext^*_\lambda$. Therefore, *MEK* is relatively consistent to $HRRC^*$.

VI. *MEK and concept correlates.*

We now consider our two theses in section II concerning computable concepts correlates. Both of them can be expressed in terms of the logical syntax, as follows:

$AC^s. \forall^s F \exists x(F = x)$, and

$AC^c. \forall^c F \exists x(F = x)$.

Clearly, thesis AC^s assigns correlates to every semi-computable concept and AC^c to every fully computable concept. When AC^s is added to the axiomatic base of *MEK*, the resulting axiomatic system can (by Th15, Th.11, A10, A12, Th.14 and UG^c prove AC^c as a theorem. This new system can also be shown to be relatively consistent to Cocchiarella's system $HRRC^* + Ext^*_\lambda$, by the same procedure employed above for *MEK*.

Now, AC^s and AC^c do not express our interpretation of correlates of computable concepts as *T*-machines. In order to construct formulas in which such an interpretation is explicitly stated, we shall consider an applied form of *MEK* to the language of arithmetic. We shall show that it is possible to construct a wff of such a language which (on the intended interpretation of the language) asserts that every semi-computable concept has (the Gödel number of) a Turing machine as its correlate. We shall also show the consistency of the formula with *MEK*[2].

Let L_{Ar} be the language containg the monadic predicate constants "*O*" and "*N*", the dyadic predicate constant "*S*" and the triadic predicate constants "*Ad*" and "*Mult*". We intend to interpret such constants, respectively, as the concepts of being a zero, being a natural number, being the succesor of a number, being the addition of two natural numbers and being the prod-

2 We are grateful to the referee for calling out attention to the problem whether it is possible of explicitly expressing our interpretation of correlates (of computable concepts) in a thesis consistent with *MEK*.

uct of two natural numbers. So L_{Ar} on the intended intuitive interpretation is a language for arithmetic.

We proceed now to recursively define the expression $N_m(x)$, which can be intuitively thought as expressing the concept of *being the natural number m* as well as to define the expression $\exists! x\varphi$ which can be read as "there is a unique x such that φ":

Definition I

$$N_o =_{df.} O$$

$$N_{m+1} =_{df.} \left[\lambda x(\exists y(S(x,y) \& N_m(y)))\right]$$

Definition II

Where φ is a wff containing x free and y is free for x in φ,

$$\exists! x\varphi =_{def.} \exists x(\forall y(\varphi y/x \leftrightarrow x = y))$$

Where *a, b, c, d* and *t* are terms of L_{Ar}, let Q be the following set of wffs of L_{Ar}:

B1. $N(a) \rightarrow \exists y(y = a)$ (provided y does not occur free in a)
B2. $N(t) \rightarrow \exists! xS(x,t)$ (provided x does not occur free in t)
B3. $(S(a,c) \& N(c)) \rightarrow N(a)$
B4. $(S(a,d) \& S(c,b)) \rightarrow (a = c \rightarrow d = b)$
B5. $\exists y S(a,y) \rightarrow \sim O(a)$
B6. $(N(a) \& \sim O(a)) \rightarrow \exists y(S(a,y))$ (provided y does not occur free in a)
B7. $(N(b) \& N(c)) \rightarrow \exists! xAd(x,b,c)$ (provided x does not occur free in both c and b)
B8. $((O(b) \& N(a)) \rightarrow Ad(a,a,b))$
B9. $S(d,c) \rightarrow (Ad(a,b,d) \leftrightarrow \exists w(S(a,w) \& Ad(w,b,c)))$ (provided w does not occur free in *c, a* and *b*)
B10. $(N(b) \& N(c)) \rightarrow \exists! xMult(x,b,c)$ (provided x does not occur free in both c and b)
B11. $(O(a) \& N(b)) \rightarrow Mult(a,b,a)$
B14. $S(c,d) \rightarrow (Mult(a,b,c) \leftrightarrow \exists w(Ad(a,b,w) \& Mult(w,b,d)))$ (provided x does not occur free in d, a and b)
B15. $(Ad(a,b,c) \rightarrow N(a))$
B16. $(Mult(a,b,c) \rightarrow N(a))$
B17. $(O(a) \rightarrow N(a))$
B18. $\exists! xO(x)$

This set of wffs constitutes our adaptation of *Tarski-Motowski-Robinson's* set of axioms of first order system Q for arithmetic to the logical syntax of this paper (which does not contain function symbols) and to the peculiarity (of being free of existential pressupositions with respect to singular terms) of the system we shall use as underlying logic (viz. *MEK*). Let *Q-MEK* be the system resulting of the addition of *Q* to the axiomatic base of *MEK*. Concerning this system, we should first note that it can easily be shown by induction, definition I, B17, B3 that, if n is a natural number and t a term of L_{Ar},

$$\vdash_{Q-MEK} N_n(t) \rightarrow N(t)$$

and so, by induction as well, B2, B18, definition I, that for every natural number n,

$$\vdash_{Q-MEK} \exists! y N_n(y).$$

On the other hand, it can also be shown (by a strategy similar to Boolos & Jeffrey's (pp. 166-68)) that every recursive function is representable in *Q-MEK*, i.e., for every recursive function f and natural numbers $n_1, \ldots, n_m, k$ there is a wff φ containing $m+1$ (distinct) variables $z, x_1, \ldots, x_m$ such that for any term $t_1 \ldots t_m, u$, if $f(n_1, \ldots, n_m) = k$, then

$$\vdash_{Q-MEK} \left(N_{n_1}(t_1) \& \ldots \& N_{n_m}(t_m) \& N_k(u)\right) \rightarrow \varphi(u/z, t_1/x_1 \ldots t_m/x_m) \text{ and}$$
$$\vdash_{Q-MEK} N_{n_1}(t_1) \& \ldots \& N_{n_m}(t_m) \& N_k(u) \rightarrow \forall x\left(\varphi(x/z, t_1/x_1 \ldots t_m/x_m) \rightarrow x = u\right).$$

provided $u, t_1 \ldots t_m$ are free (respectively) for $z, x_1, \ldots, x_m$ in φ. By a *recursive predicate* we shall mean an expression $P(x_1 \ldots x_n)$ containing as free variables only $x_1 \ldots x_n$ (which range over the natural numbers) and such that the characteristic function of the set $\{<n_1, \ldots, n_n> \in \omega^n : P(n_1, \ldots, n_n)\}$ is recursive. Since *Q-MEK* is strong enough to represent every recursive function, then clearly monadic recursive predicates are representable in *Q-MEK* as well. That is, more precisely, for every monadic recursive predicate P and natural number n, there is a wff σ of *Q-MEK* containing one free variable x such that, for every term t of L_{Ar} free for x in σ,

if $P(n)$ is true, then $\vdash_{Q-MEK} N_n(t) \rightarrow \sigma(t/x)$
if $P(n)$ is false, then $\vdash_{Q-MEK} N_n(t) \rightarrow \sim \sigma(t/x)$

Now, let $TM(x)$ be the predicate defined in Davis (p.60) which, for any natural number m, $TM(m)$ holds if and only if m is the Gödel number of a Turing machine (under the arithmetization of Davis (pp. 56-7). As shown in Davis (ibid.), TM is a monadic primitive recursive predicate and so it is representable in *Q-MEK*. That is, there is a monadic predicate expression φ of *Q-MEK* containing one free variable y such that, for every term t of L_{Ar} free for y in φ,

if $TM(n)$ is true, then $\vdash_{Q-MEK} N_n(t) \rightarrow \varphi(t/y)$, and
if $TM(n)$ is false, then $\vdash_{Q-MEK} N_n(t) \rightarrow \sim \varphi(t/y)$.

Assume now an arithmetization of the syntax of *Q-MEK*. Let T be the predicate expression containing one free variable, with the smallest Gödel number under such an arithmetization, representing *TM* in *Q-MEK*. We are now able to fully express our interpretation of computable concepts correlates, as follows:

$$T^s.(\forall^s F)(\exists y)(T(y) \& y = F)$$
$$T^c.(\forall^c F)(\exists y)(T(y) \& y = F)$$

The reader can easily verify that T^c follows from T^s within the context of *Q-MEK* [see p. 352].

We proceed now to show the consistency of T^s with *MEK*. Thus, we shall be showing that the interpretation of semi-computable concepts correlates as T-machines is consistent with the logic of *MEK* together with the assumption that every semi-computable concept has a correlate. Now, before proceeding to state the consistency proof, we shall first briefly outline it.

We shall begin proving the consistency of T^s with respect to system *MEK* by firstly developing a formal semantics. System *MEK* will turn out to be sound with respect to this semantics, i.e., every theorem of *MEK* will be valid in every model of the semantics. Now, we should note that we shall not be claiming that the semantics is philosophically adequate for the intended interpretation of the epistemic operator. Its role will be rather instrumental in proving consistency of *MEK* + T^s. We should also point out that we have constructed the semantics by modifying Cocchiarella's intensional Fregean-semantic system in Cocchiarella (1986a, chap.VI).

Next, we shall proceed to construct a model belonging to the semantics (and which we shall call "*H*") in which every formula of *Q* will valid. Clearly, by soundness of *MEK* with respect to the semantics, *H* will also constitute a model of *Q-MEK*, i.e., every theorem of *Q-MEK* will be valid in the model.

For every n-place predicate variable F^n, the range of "$\forall^s F^n$" in *H* will be the set of all ordered pairs $< 1, S >$ such that S is a *semi-computable* subset of n-tuples of natural numbers. Model *H* will also contain a function g

assigning to every member $< 1, S >$ of the range of "$\forall^s F^n$" (in H) the gödel number of a T-machine computing S. (Definition of this function will be based on results from computability theory). Intuitively speaking, function g will (set-theoretically) represent in H the correlation of every n-ary computable-concept in H (i.e., every member of the range of "$\forall^s F^n$" in H) with T-machines. Consequently, relative to H, every n-ary semi-computable concept (in H) will have a correlate, viz., the gödel number of a T-machine. On the basis of this and other features of H, the fact that being (the gödel number of) a T-machine is a recursive property (and consequently representable in *Q-MEK*) and the validity of *Q-MEK* in H, we shall be able to show the validity of T^s in H. Therefore, since both *Q-MEK* and T^s are valid in H, the consistency of T^s with respect to *MEK* will clearly follow.

Consistency of Q-MEK + T^s

By a modified Cocchiarellan frame (C-frame, for short) we shall understand a structure $< D, S_n, Y_n, f_i >_{n \in \omega, i \in W}$, where ω is the set of natural numbers and such that (1) D and W are non-empty sets; (2) for all $n \in \omega, S_n \subseteq Y_n \subseteq \mathbb{P}(D^n)^W$, where "$\mathbb{P}(D^n)^W$" stands for the set of functions from W into the power set of D^n (for $n = 0$, we set $D^0 = \{\phi\}$); and (3) there is a set D^* such that $D \subset D^*$and, for $i \in W$, f_i is a function from $D^* \cup \left(\bigcup_{n \in \omega} Y_n\right)$ into D* such that

(i) for all $d \in D^*, f_i(d) = d$,

(ii) for every $z \in \bigcup_{n \in \omega} S_n$, there is a $d \in D$ such that $f_i(z) = d$, and

(iii) for every $n \in \omega, f_i$ restricted to Y_n is one to one.

Set D is the domain of discourse as well as the range of values of the bound individual variables. Set D^* is the range of values of the free individual variables. Sets S_n and Y_n are, respectively, the range of values of the n-place variables bound by the quantifier "$\forall^s$" and the range of values of the n-place variables bound by the universal quantifier "$\forall$". For $i \in W, f_i$ set-theorically represents the correlation relative to i of a n-ary concept with an object.

Where $\mathcal{U}$ is a C-frame, we shall say that A is an assignment (of values to variables in $\mathcal{U}$) if A is a function with the set of variables as domain and such that (1) for all $n \in \omega$, all n-place predicate variables F^n, $A(F^n) \in \mathbb{P}(D^n)^W$ and (2) for each individual variable x, $A(x) \in D^*$. Also we set $A(d / u) = (A - \{< u, A(u) >\} \cup \{< u, d >\})$, i.e., $A(d / u)$ is that referential assignment which is exactly like A except (at most) for its assigning d to u. Now, where L is a language and $\mathcal{U}$ a C-frame, we say that $I = < h, \mathcal{U} >$ is a (modified) Cocchiarellan intensional model for L (L -C model, for short), if h is a function with L as domain such that for all $n \in \omega$ and all n-place predicate constants $P \in L, h(P) \in \mathbb{P}(D^n)^W$ and for each individual constant $c \in L, h(c) \in D^*$.

If σ is a meaningful expression of L, i.e., $\sigma \in ME(L)$, and $I = < g, \mathcal{U} >$ is a L-C model and A an assignment in $\mathcal{U}$, then we define the intension of σ in I relative to A (in symbols, $int_{I,A}(\sigma)$ as follows:

0. If a is a variable, then $int_{I,A}(a) = A(a)$. If $c \in L$ (i.e. c is a predicate or individual constant), then $int_{I,A}(c) = g(c)$.
   For every $n \in \omega$ and $\sigma \in ME_{n+1}(L)$, we recursively define the intension of σ relative to A as that function in $\mathbb{P}(D^n)^W$ such that
1. If σ is $\pi(a_1 \ldots a_n)$, where $\pi \in ME_{n+1}$ and $a_1 \ldots a_n \in ME_0$, then for all $i \in W$,
   $int_{I,A}(\sigma)(i) = 1$ iff $< f_i(int_{I,A}(a_1)) \ldots f_i(int_{I,A}(a_n)) > \in int_{I,A}(\pi)\ (i)$;
2. If σ is $a = b$, where $a, b \in ME_0$ (L), then for all $i \in W$, $int_{I,A}(\sigma)(i)$ $= 1$ iff $f_i(int_{I,A}(a)) = f_i(int_{I,A}(b))$;
3. If σ is $[\lambda x_1 \ldots x_n \varphi]$, where $\varphi \in ME_1(L)$, then for all $i \in W$,
   $int_{I,A}(\sigma)(i) = \{< d_1, \ldots, d_n > \in D^n : int_{I,A(d_1/x_1 \ldots d_n/x_n)}(\varphi)(i) = 1\}$
4. If σ is $\sim \varphi$, where $\varphi \in ME_1(L)$,, then for all $i \in W$, $int_{I,A}(\sigma)(i) = 1$ iff $int_{I,A}(\varphi)(i) = 0$
5. If σ is $[e]\varphi$, where $\varphi \in ME_1(L)$,, then for all $i \in w$, $int_{I,A}(\sigma)(i) = 1$ iff for all $k \in W$, $int_{I,A}(\varphi)(k) = 1$
6. If σ is $(\varphi \rightarrow \gamma)$, where $\varphi, \gamma \in ME_1(L)$, then for all $i \in W$, $int_{I,A}(\sigma)(i) = 1$ iff either $int_{I,A}(\varphi)(i) = 0$ or $int_{I,A}(\gamma)(i) = 1$
7. If σ is $\forall x \gamma$, where $\gamma \in ME_1(L)$, then for all $i \in W$, $int_{I,A}(\sigma)(i) = 1$ iff for all $d \in D$, $int_{I,A(d/x)}(\gamma)(i) = 1$
8. If σ is $\forall F^n \gamma$, where $\gamma \in ME_1(L)$, then for all $i \in W$, $int_{I,A}(\sigma)(i) = 1$, iff for all $d \in Y_n$, $int_{I,A(d/F)}(\gamma)(i) = 1$
9. If σ is $\forall^s F^n \gamma$, where $\gamma \in ME_1(L)$, then for all $i \in W$, $int_{I,A}(\sigma)(i) = 1$ iff for all $d \in S_n$, $int_{I,A(d/F)}(\gamma)(i) = 1$
10. If σ is $[\lambda \varphi]$, where $\varphi \in ME_1(L)$,, then for all $i \in W$, $int_{I,A}(\sigma)(i) = int_{I,A}(\varphi)(i)$

Where I, $\mathcal{U}$ and A are as above, and $i \in W$, we define satisfaction, truth and validity of a wff of L as follows:

(i) A satisfies φ in I at i iff $int_{I,A}(\varphi)(i)=1$;
(ii) φ is true at i in I iff every assignment in $\mathcal{U}$ satisfies φ in I at i; and
(iii) φ is valid in I iff for every $i \in W$, φ is true at i in I.

The reader can easily verify that *MEK* is sound with respect to the above semantics, i.e., if $\vdash_{MEK}\varphi$, then φ is valid in every L-C model, for any wff φ of L. In the case of axioms A5 and A6, the reader should proceed by strong induction on the complexity of φ. We should note that clause (iii) of page 356 plays an important role in this inductive proof. Validity of A10 follows from clauses 2 and 8-9. With respect to A13, the reader should proceed by strong induction on the complexity of σ. Concerning A16-18, it should be taken into account that their respective domains of quantification do not vary across all $i \in W$. We should point out that, due to clause (ii) of page 356, AC^s is also valid and so $MEK + AC^s$ is sound with respect to the above semantics as well.

We now proceed to construct a $L_{Ar} - C$ model in which every theorem of *Q-MEK* + T^s is valid, showing then, at the same time, the consistency of *MEK* + T^s. However, we shall first state results formulated and proved in Davis, which we shall employ in the construction of the model.

Let us consider the set of first order well formed formulas of the language whose set of non-logical symbols are ·, +, ' (which are to be interpreted, respectively, as the multiplication , addition and succesor functions) and 0 (which is to be assigned the number zero as denotation). By a *numerical predicate* we shall mean any of such well formed formulas containing free variables. Clearly, numerical predicates qualify as predicates in the sense of Davis (p.xxii). So, by Davis (p. 66), a numerical predicate $P(x_1 \ldots x_n)$ is *semicomputable* if and only if there is a partially computable function whose domain is the set $\{<n_1 \ldots n_n> \in \omega^n : P(n_1 \ldots n_n)\}$. [Briefly, by a *partially computable function* it is understood a partial numerical function computable by a T-machine (cf. Davis (p.10)]. Now, by a *semicomputable set* S^n we shall mean a set of n-tuples of natural numbers for which there is a semicomputable numerical predicate $P(x_1 \ldots x_n)$ such that

$$S^n = \{<n_1 \ldots n_n> \in \omega^n : P(n_1 \ldots n_n)\}$$

By the so called *Klenee's enumeration theorem* [cf Davis (p.67, theorem 1.4)], for every semicomputable numerical predicate $P(x_1 \ldots x_n)$ there is a natural number z such that z is the gödel number of a T-machine and

$$P(x_1 \ldots x_n) \leftrightarrow (\exists y) T_n(z, x_1 \ldots x_n, y)$$

where the predicate $T_n(z, x_1 \dots x_n, y)$ is defined as " z is the Gödel number of a Turing machine Z, y is the Gödel number of a computation with respect to Z having only the (Turing representation) of $x_1 \dots x_n$ on the tape in its initial state ". We should note that such a predicate is primitive recursive and, moreover, that if $T_n(z, x_1 \dots x_n, y)$, then $TM(z)$ [cf. Davis (pp. 57, 62)] and so (since T represents TM in *Q-MEK*), for every term t of $L_{Ar'}$, $\vdash_{Q\text{-}MEK} N_z(t) \rightarrow T(t)$. On the other hand, by definition of a semi-computable set and Kleene's enumeration theorem, for every semicomputable set S^n, the set

$$G_{S^n} = \{z \in \omega\text{: for every } x_1 \dots x_n \in \omega, < x_1 \dots x_n > \in S^n \leftrightarrow (\exists y) T_n (z, x_1 \dots x_n, y) \text{ and } z \text{ is the gödel number of a T} - \text{machine}\}$$

is not empty. Let L_{S^n} be the least element of G_{S^n}. By above remark, it follows that, since $TM(L_{S^n})$, i.e. L_{S^n} is the gödel number of a T-machine, for every term t, $\vdash_{Q\text{-}MEK} N_{L_{S^n}}(t) \rightarrow T(t)$

We can now proceed to construct a model for *Q-MEK* + T^s. Let ω be the set of natural numbers,

$$D^* = \omega \cup \bigcup_{n \in \omega} \{A \subseteq \omega^n : A \text{ is not semicomputable}\}$$

W = {1} and $\mathfrak{B}$ be the following structure:

$< \omega, S_n, C_n, g >_{n \in \omega, W}$, where

1) $S_n = \{<1, A>: A \in \mathbb{P}(\omega^n) \text{ and } A \text{ is semicomputable}\}$,
2) $C_n = \{<1, B>: B \in \mathbb{P}(\omega^n)\}$,
3) g is the function from $D^* \cup (\bigcup_{n \in \omega} C_n)$ into D^* such that

   (i) for all $d \in D^*$, $g(d) = d$,
   (ii) for every $n \in \omega$, if $z =< 1, B > \in S_n$ (i.e. B is a semicomputable subset of ω^n), then $g(z) = L_B$ (i.e., the least element of G_B) and
   (iii) if $z =< 1, A > \in (\bigcup_{n \in \omega} C_n) - (\bigcup_{n \in \omega} S_n)$, $g(z) = A$.

Clearly, $\mathfrak{B}$ is a C-frame. Let $H = < h, \mathfrak{B} >$, where h is a function with L_{Ar} as domain such that:

$$h(N) = \{<1, \omega>\},$$
$$h(Add) = \{<1, +>\},$$
$$h(Mult) = \{<1, x>\},$$
$$h(O) = \{<1, \{0\}>\},$$
$$h(S) = \{<1, s>\},$$

where $+ = \{<a, b, c> \in \omega^3$: b plus c equals $a\}$, $x = \{<a, b, c> \in \omega^3$: a is the product of b and $c\}$, $s = \{<a, b> \in \omega^2$: a is the sucessor of $b\}$.

H clearly is a L_{Ar} -*C* model and so, by soundness, *MEK* is valid in *H*. Also, as the reader can easily verify, every member of *Q* is valid in *H*. Therefore, if $\vdash_{Q\text{-}MEK} \varphi$, then φ is valid in *H*. Now, since *Q-MEK* is valid in *H* and for every natural number *n* and term *t* of L_{Ar}

if $TM(n)$ is true, then $\vdash_{Q\text{-}MEK} N_n(t) \rightarrow T(t)$ and
if $TM(n)$ is false, then $\vdash_{Q\text{-}MEK} N_n(t) \rightarrow\sim T(t)$

then

if $TM(n)$ is true, then $(N_n(t) \rightarrow T(t))$ is valid in *H*, and
if $TM(n)$ is false, then $(N_n(t) \rightarrow\sim T(t))$ is valid in *H*.

On the other hand, it can be shown by induction on *n* that

if *t* is a term of L_{Ar}, *n* a natural number, *A* an assignment in *H* and $i \in W$, if *A* satisfies $N_n(t)$ in *H* at *i*, then $g(int_{H,A}(t)) = n$

The validity of T^s in *H* now follows from these results, the definition of the correlation function [see clause 3ii], semantic clauses concerning quantifiers and identity, and (since $\vdash_{Q\text{-}MEK} \exists! yN_n(y)$) the validity of $\exists! yN_n(y)$ in *H*.Therefore, since both *Q-MEK* and T^s are valid in *H*, then T^s is consistent with Q-*MEK*.

VII. *Existential presuppositions*

As noted above, *MEK* is free of existential presuppositions concerning predicate expressions. Although several principles can be formulated imposing such presuppositions, we shall only pay attention to those directly related to the two major forms of concept-formation, *viz., holistic conceptualism and constructive conceptualism,* and consider the problem whether these principles can be consistently added to axiomatic base of system

$MEK + AC^s$, i.e. MEK together with the assumption that every semi-computable concept has a correlate. We shall also consider the problem whether those same principles can be consistently added to the system $Q\text{-}MEK + T^s$ (in which our interpretation of computable concepts correlates as T-machines is explicitly stated).

Holistic conceptualism is the widest form permitted of concept-formation. It maintains that a stage of concept-formation can be reached at which every predicate expression of a given language will stand for a concept. This sort of conceptualism also allows for impredicative concept-construction, that is, formation of concepts presupposing a totality to which they belong. In contradistinction to holistic conceptualism, concepts formed in accord with the principles of constructive conceptualism are predicative, that is, their formation do not presuppose totalities to which they belong. Consequently, this form of concept construction obeys Poincare-Russell vicious circle principle interpreted as a principle for the introduction of concepts. We should note that holistic conceptualism is not opposed to constructive conceptualism, since it presupposes formation of predicative concepts as basis for the process which leads to impredicative concept-formation. However, contrary to holistic conceptualism, a strict form of constructive conceptualism will reject impredicative concepts and, consequently, for this variant of conceptualism only predicative predicate expressions could stand for concepts. [For details on holistic and constructive conceptualism, see Cocchiarella (1986b)]

The above two major variants of conceptualism philosophically validate principles (imposing existential presuppositions) which, as the reader will note, amount to forms of comprehension schemata, the most general of which is $\exists F(F=[\lambda x_1 \ldots x_n \varphi])$ (where F does not occur free in φ). This schema, known in Cocchiarella(1986a) as CP^*, can be obviously justified within the philosophical framework of holistic conceptualism. It should be noticed that we will obtain a principle expressing a version of impredicative conceptualism in which the identity sign does not stand for a concept, by prescribing that φ in CP^* should not contain the identity sign. This schema corresponds to a form of holistic conceptualism which does not look at identity as a concept at all and at the identity sign as a predicate expression. Following Cocchiarella(1986a), we shall call this new principle CP^{**}. Now, a comprehension schema corresponding to constructive conceptualism can be stated as : $\forall G_1 \ldots \forall G_n \exists F(F=[\lambda x_1 \ldots x_n \varphi])$, where $[\lambda x_1 \ldots x_n \varphi]$ is a lambda abstract in which (1) neither F, the identity sign, nor any predicate constant occurs (2) no predicate variable has a bound occurrence, and (3) $G_1 \ldots G_n$ are all of the predicate variables occurring (free) in φ. Again, following Cocchiarella(1986a), we shall refer to this schema as CCP^*.

By the same procedure employed in section VI to show the relative consistency of $MEK + AC^s$, we can also show the relative consistency of systems $MEK + AC^s + CP^*$, $MEK + AC^s + CCP^*$ and $MEK + AC^s + CP^{**}$ to system $HRRC^* + Ext^*_\lambda$. On the other hand, both $Q\text{-}MEK + T^s + CP^*$ and $Q\text{-}MEK + T^s + CCP^*$ can also be shown to be consistent systems: given that we have shown in section VI that $Q\text{-}MEK + T^s$ is valid in H, it is sufficient to note that both CP^* and CCP^* are valid schemata in H as well, since the universal quantifier applied to n-place predicate variables ranges in H, at every $i \in W$, over the entire power set of ω^n. Therefore, the philosophical framework pressuposed by both $MEK + AC^s$ and $Q\text{-}MEK + T^s$ is logically coherent with the two important ways of looking at concept-formation represented by schemata CP^* and CCP^*.

We shall now consider the possibility that, relative to a given universe of discourse U and language L for this universe, a conceptualization of U would be such that every possible predicate expression of L would stand for a computable concept. This possibility would constitute one in which the holistic approach to concept formation applies to computable concepts and so a situation relative to which $\exists^s F(F = [\lambda x_1 \ldots x_n \varphi])$ and/or $\exists^c F(F = [\lambda x_1 \ldots x_n \varphi])$ (where F does not occur free in φ and φ is a wff of L) will turn out to be valid. Let us refer to these schemata as CP^s and CP^c, respectively. Now, in this situation it would not be possible to reason in accordance with the logic of $MEK + AC^s$ without falling into a contradiction, that is, neither CP^s nor CP^c can be consistently added to the axiomatic base of $MEK + AC^s$. Both schemata will respectively have $\exists^s F(F = [\lambda x_1 (\exists^s G)(G = x_1 \,\&\, \sim G(x_1))])$ and $\exists^c F(F = [\lambda x_1 (\exists^c G)(G = x_1 \,\&\, \sim G((x_1))])$ as their instances, from which an inconsistency can be derived due to the presence of AC^s as well as of AC^c in such a system [see above]. More precisely, it can be shown, on one hand, that both

I. $$\forall y(y = [\lambda x_1 \exists^s G(G = x_1 \,\&\, \sim G(x_1))] \rightarrow$$
$$([\lambda x_1 \exists^s G(G = x_1 \,\&\, \sim G(x_1))](y) \rightarrow$$
$$\sim [\lambda x_1 \exists^s G(G = x_1 \,\&\, \sim G(x_1))](y)))$$

and

II. $$\exists^s G(G = [\lambda x_1 \exists^s G(G = x_1 \,\&\, \sim G(x_1))] \rightarrow$$
$$(\forall y(y = [\lambda x_1 \exists^s G(G = x_1 \,\&\, \sim G(x_1))] \rightarrow$$

$$\left(\sim\left[\lambda x_1 \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right](y) \rightarrow \right.$$
$$\left.\left[\lambda x_1 \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right](y)\right)\right)$$

are theorems of $MEK + AC^c$:

Proof of I :

1. $\left[\lambda x_1 \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right](y) \rightarrow \exists x_1\left(x_1 = y \& \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right]\right)$
   (by A13)
2. $\exists x_1\left(x_1 = y \& \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right] \rightarrow \exists^s G\left(G = y \& \sim G(y)\right)$
   (by A6, PL, U.G., A2 and A4)
3. $\exists^s G\left(G = y \& \sim G(y)\right) \rightarrow (y = \left[\lambda x_1 \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right] \rightarrow$
   $\sim\left[\lambda x_1 \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right](y))$
   (by A6, PL, UG^S, A3 and Th. 2.)
4. $\forall y(y = \left[\lambda x_1 \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right] \rightarrow \left(\left[\lambda x_1 \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right](y) \rightarrow\right.$
   $\left.\sim\left[\lambda x_1 \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right](y)\right))$
   (from 1-3 by PL and UG)

Proof of II :

1. $\exists^s G\left(G = \left[\lambda x_1 \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right]\right) \rightarrow$
   $\left(\left(\left[\lambda x_1 \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right] = y\right) \rightarrow\right.$
   $\left(\sim\left[\lambda x_1 \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right](y) \rightarrow \exists^s G\left(G = y \& \sim G(y)\right)\right)\right)$
   (by Th.8 and PL)
2. $\left(\exists x(x = y) \& \exists^s G\left(G = y \& \sim G(y)\right)\right) \rightarrow \exists x\left(x = y \& \exists^s G\left(G = x \& \sim G(x)\right)\right)$
   (by A1, A14 Th.6 and PL)
3. $\exists x\left(x = y \& \exists^s G\left(G = x \& \sim G(x)\right)\right) \rightarrow \left[\lambda x_1 \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right](y)$
   (by A13)
4. $\exists^s G\left(G = \left[\lambda x_1 \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right]\right) \rightarrow$
   $\forall y\left(\left(y = \left[\lambda x_1 \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right]\right) \rightarrow\right.$
   $\left(\sim\left[\lambda x_1 \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right](y) \rightarrow \left[\lambda x_1 \exists^s G\left(G = x_1 \& \sim G(x_1)\right)\right](y)\right)\right)$
   (From 1-3, by PL, UG, A2, A4 and A9)

On the other hand, when CP^s is added to the axiomatic base of MEK + AC^s , then the following are theorems of $MEK + AC^s + CP^s$:

III. $\forall y\big(y = [\lambda x_1 \exists^s G(G = x_1 \,\&\, {\sim} G(x_1))] \rightarrow$
$({\sim}([\lambda x_1 \exists^s G(G = x_1 \,\&\, {\sim} G(x_1))](y)) \rightarrow$
$[\lambda x_1 \exists^s G(G = x_1 \,\&\, {\sim} G(x_1))](y))$
(by II above, CP^s, and PL),

IV. $\exists y\big(y = [\lambda x_1 \exists^s G(G = x_1 \,\&\, {\sim} G(x_1))]$;
(by CP^s, AC^s, Th.8 and PL)

V. ${\sim}[\lambda x_1 \exists^s G(G = x_1 \,\&\, {\sim} G(x_1))]([\lambda x_1 \exists^s G(G = x_1 \,\&\, {\sim} G(x_1))]) \leftrightarrow$
$[\lambda x_1 \exists^s G(G = x_1 \,\&\, {\sim} G(x_1))]([\lambda x_1 \exists^s G(G = x_1 \,\&\, {\sim} G(x_1))]))$
(by I, III, IV, Th. 6, A1 and PL)

But then $MEK + AC^s + CP^s$ is inconsistent, since by PL

VI. ${\sim}([\lambda x_1 \exists^s G(G = x_1 \,\&\, {\sim} G(x_1))]([\lambda x_1 \exists^s G(G = x_1 \,\&\, {\sim} G(x_1))]) \leftrightarrow$
${\sim}[\lambda x_1 \exists^s G(G = x_1 \,\&\, {\sim} G(x_1))]([\lambda x_1 \exists^s G(G = x_1 \,\&\, {\sim} G(x_1))]))$

is also a theorem of $MEK + AC^s + CP^s$. An analogous inconsistency for the system $MEK + AC^s + CP^c$ can be obtained by both replacing use of CP^s and AC^s, in the proofs of I-VI, by CP^c and AC^c respectively, and by replacing "$\exists^s F$" wherever it occurs in I-VI by "$\exists^c F$". Finally, we should point out that similar contradictions would follow from either Q-$MEK + T^s + CP^c$ or Q-$MEK + T^s + CP^s$ since in both of these system AC^s can be shown to be a theorem.

We have above taken into account a possible relationship between the holistic approach to concept formation and computable concepts. We shall now consider the problem whether computable concept formation can be carried out in accordance with the principles of constructive or predicative conceptualism. The following two schemata would represent this possibility:

(CCP^s): $(\forall^s G_1)\ldots(\forall^s G_n)(\exists^s F)(F = [\lambda x_1 \ldots x_n \varphi])$, *where* $[\lambda x_1 \ldots x_n \varphi]$ *is a lambda abstract in which* (1) *neither F, the identity sign, nor any predicate constant occurs* (2) *no predicate variable has a bound occurrence, and* (3) $G_1 \ldots G_n$ *are all of the predicate variables occurring (free) in* φ.

(CCP^c): $(\forall^c G_1)\ldots(\forall^c G_n)(\exists^c F)(F = [\lambda x_1 \ldots x_n \varphi])$, *where* $[\lambda x_1 \ldots x_n \varphi]$ *is a lambda abstract in which* (1) *neither F, the identity sign, nor any predicate constant occurs* (2) *no predicate variable has a bound occurrence, and* (3) $G_1 \ldots G_n$ *are all of the predicate variables occurring (free) in* φ.

Now, neither (CCP^s) nor (CCP^c) will be valid if we take into account both our discussion and assumptions in the previous sections of this article together with certain results springing from computability theory. An important consequence of those sections is that, if *P* is a predicate expression, *P* is Turing-computable (partially Turing-computable) if and only if a fully (semi-) computable concept could be formed for which *P* will stand. Given this consequence (which hereafter we shall call *Cn*), it is possible to find invalid instances of both schemata. In the case of (CCP^s), there is the formula $\forall^s F \exists^s G(G = [\lambda y \sim F(y)])$ which implies that the negation of a predicate expression (standing for a semi-computable concept) will also stand for a semi-computable concept. Now, since there is a partially Turing-computable predicate expression *P* whose negation is not a partially Turing-computable predicate [cf. Davis (p. 68)], then by *Cn* a semi-computable concept can be formed for which *P* will stand and no semi-computable concept could be formed for which the negation of *P* would ever stand. Thus, (CCP^s) has a false instance and cannot be intuitively valid.

On the other hand, concerning (CCP^c) we should first note that, according to computability theory, there is a predicate expression *P* springing from the addition of the existential quantifier to a certain two-place Turing-computable predicate $R(x, y)$ which is not Turing-computable; so, *P* would be $[\lambda x \exists y R(x, y)]$ (see Davids (pp. 62,66-68) for a description of this predicate and a proof that is not Turing computable). Consequently, by *Cn*, since $[\lambda x \exists y R(x, y)]$ is not Turing-computable, the formula $\exists^c G(G = [\lambda x \exists y R(x, y)])$ would be false. However, this latter formula follows from (CCP^c), *Cn* and the Turing-computability of the predicate expression *R*, and so (CCP^c) cannot be valid.

It is important to note that, due to the above remarks concerning constructive or predicative concept formation as applied to computable concepts, neither semi-computable nor fully computable concepts can be identified with Cocchiarella's (1986b) ramified constructive concepts. This is because Cocchiarella's principle characterizing ramified constructive concept formation, viz.:

$(RCCP!^{*}_{\lambda})\forall^j x_1 \ldots \forall^j x_n \forall^j F_1 \ldots \forall^j F_n \exists^j G(G = [\lambda y_1 \ldots y_k \varphi])$, for every $j \in \omega$ - {0}, (where φ should satisfy certain conditions stated in Cocchiarella (op. cit.)), will have , at every $j \in \omega$ - {0}, the above mentioned formulas among its instances. Also, since neither semi-computable nor fully computable concepts can be identified with Cocchiarella's constructive concepts, the epistemic operator "it is constructibly knowable that p", introduced in Cocchiarella (op.cit.) and which we have intuitively interpreted and formally investigated in Freund (1991), is different from the epistemic operator we have considered in this article.

Departamento de Filosofia,
Universidad Nacional APDO 86-3000 Heredia, Costa Rica

REFERENCES

(1) G.Boolos & R. Jeffry (1980), "Computability and Logic", *Cambridge University Press.*

(2) Nino B. Cocchiarella (1986a), "Logical Investigations of Predication Theory and the Problem of Universals", *Bibliopolis Press*, Naples.

(3) Nino B. Cocchiarella (1986b), "Conceptualism, Ramified Logic and Nominalized Predicates", *Topoi* 5, pp. 75-87.

(4) Nino B. Cocchiarella (1989), "Conceptualism, Realism and Intensional Logic", *Topoi* 8, pp. 15-34.

(5) Nino B. Cocchiarella (1993), "Conceptual Realism as a Formal Ontology", Ms.

(6) M. Davis (1958), "Computability and Unsolvability", *Cambridge University Press.*

(7) Max A. Freund, (1991), "Consideraciones logico-epistemicas relativas a una forma de conceptualismo ramificado", *Critica* 69, pp. 3-27.

(8) S. C. Kleene (1952), "Introduction to Metamathematics", *North Holland Publishing Co.*, Amsterdam.

(9) W. N. Reinhardt (1985), "Absolute Versions of Incompleteness Theorems" *Nous,* 19, pp. 317-45.

(10) W. N. Reinhardt (1986), "Epistemic Theories and the Interpretation of Gödel's Incompleteness Theorems", *Journal of Philosophical Logic,* 15, pp. 427-74.

(11) S. Shapiro (1985), "Intensional Mathematics", *North Holland Publishing Co.*, Amsterdam.