Notre Dame Journal of Formal Logic
Volume XX, Number 1, January 1979
NDJFAM

THEORY OF OBJECTS AND SET THEORY: INTRODUCTION AND SEMANTICS

ANDRÉ CHAUVIN

1 *Preliminary* By Kleene's *Introduction to Metamathematics* (1952), CH. XIV, §72, Theorems 35 and 38, *if consistent*, a first order theory has "*Kleene's models*", i.e., models where the universe is the set of natural integers 0, 1, . . ., n, . . . and the fundamental relations are each of the form $P(a_1, \ldots, a_n)$ with

$$P(a_1, \ldots, a_n) \leftrightarrow (\exists x)(\forall y) R(a_1, \ldots, a_n, x, y) \leftrightarrow (\forall x)(\exists y) S(a_1, \ldots, a_n, x, y)$$

where R and S are primitive recursive and "$\leftrightarrow$" is the biconditional.

Trying to prove the consistency of first order *set theory* by finding directly such a model (which must exist if this theory is consistent), we thought, as partial recursive functions are basic notions of Theory of Algorithms, to which belong Kleene's models, that it would be better to bring *set theory* nearer to *theory of algorithms*, by taking for the first one as *primitive* the notion of *partial* function and the relation of equality and as *derived* the notion of set and the relation of membership (if we do not impose extensionality to our partial functions we are again nearer to the primitive basic notion of the theory of algorithms, say the notion of algorithm itself).

We were confirmed in this idea by reading Von Neumann's "Eine Axiomatizierung der Mengenlehre". As a matter of fact, Von Neumann takes as primitive the notion of (total) function and the relation of equality. But he stresses that his system presents some arbitrariness. *We thought that this arbitrariness lay at the very beginning of Von Neumann's system, when he divides a priori its objects* ("*Ding*") *between* I-*objects* x (arguments) *and* II-*objects* f (functions), in such a way that the domain O of Von Neumann's objects is the union of O_I, the domain of arguments, and O_{II}, the domain of functions, and when he considers that the result of the binary operation [,] of application of a *function* to an argument, say $[f,x]$, is defined on and only on the cartesian rectangle $O_{II} \times O_I$. Von Neumann admits again (implicitly) that the term $[f,x]$ has always a meaning, f being always taken in O_{II} and x in O_I, and he states that this term describes an argument (a I-object).

Received January 2, 1974

Rejecting the a priori separation of the objects (i.e., the elements of a given *a priori* domain O) *between* I-*objects and* II-*objects*, we were naturally induced to admit that we could put every object x, y into every void place in [,], but that the result $[x, y]$[1] could be *undefined* (i.e., could not describe any object but could be simply a pure meaningless string of symbols, especially because x needs not to be a function at all) and, *if defined*, could be an arbitrary object. So we arrived at a more general notion of application than the one of Von Neumann, the notion of application [,] of an arbitrary object x to an arbitrary object y, the eventual result of this application, if it exists, being the object $[x, y]$. From the binary fundamental operation [,], by fixing its first argument x, we get a unary operation $[x, \ ]$, which may have an object $[x, y]$ as value when we fill the void place with y. So the operation $[x, \ ]$ constitutes a unary function f (of which x is a kind of notation) and we suppose that our notion of application is sufficiently large for getting in this way every unary partial function.

Clearly our notion of partial function is prior to every notion of set (hence of graph),[2] and is considered as a *rule*, or, better, as a *process* prescribed by this rule, which gives an eventual value from a given argument. *If we take as primitive our general notion of application*, and as derived the notion of unary partial function, it suffices to add the Schönfinkel conception of the n-ary functions as sequences of n applications of unary functions (say $F(x, y)$ is understood as $[[f, x], y])$[3] for falling into the familiar ideas of Combinatory Logic. So was the stream of ideas which led us to the present work, where we present a reconstruction of set theory from the usual notion of application of the Combinatory Logicians.

2 *The notion of function as process and the notion of application* Historically, functions were considered *first* as *rules* (*commands*) prescribing an *activity* (a *process*) aimed to get a *result* (an *object*) when applied to some objects, or, so to say, *as this activity or process itself. Later on*, the notion of function shifted from the primitive notion of process to the set-theoretic notion of *graph*, via the "representative curves" which were used for representing "graphically" various functions, with a remarkable *uniformity* opposite the extreme diversity of the processes in a time where no general notion of process (and especially the notion of algorithm) was in view. Let us make some remarks.

2.1 Something is lost of the primitive notion of a function F (as process),

1. Frequently abbreviated as xy.

2. As is Frege's notion of Predicate.

3. At the beginning of §2 in Van Heijenoort and others translation of Schönfinkel in "From Frege to Gödel," we find that a function is defined as a correspondence giving, for every argument, *at most* a value. So Schönfinkel, contrary to Von Neumann, considers the functions as partial. Von Neumann seems to have ignored Schönfinkel's work and especially his reduction of functions to unary ones.

if we identify this function with its graph f, *forgetting* the *uniform process* which restitutes a function from its graph, say the process prescribed by the following command:

"For the object x, search if an ordered pair, with x as first element, is a point of the graph f. If it is the case, take the second element of the point as the value Fx of the function F for x" (Γ)[4]

The extreme simplicity of this uniform process may give rise to such an oversight, above all if we write fx instead of Fx for the value of F for x. But the "inert" object f is, by itself, unable to act on something. Taking in such a way the part f for the whole (f, Γ) or F, an object (a graph) which characterizes but is not a function for this function itself, would be more difficult in the case where the function is a partial recursive function F, the object which characterizes it is its Gödel number f, because the uniform process which restitutes the process F from its characterization, the "inert" Gödel number f, is more intricate than the preceding: it goes from f, through a decoding, to another "inert" object, the scheme f' of the algorithmic process F, the passage from every scheme f' to the process F it commands, being, as remarked Shanin, an essential part of the understanding of the notion of algorithm: so the algorithmists are accustomed to represent a partial recursive function F by some symbol like $\{f\}$, distinct of the Gödel number f. It is more difficult, through our deep-rooted habits, to see a graph f as something like a Gödel number of F, i.e., a simple characterization of F, but not as F itself.

This simple remark throws some light upon the so called "self-application", i.e., the application of a function to itself. Clearly it is not the graph f that applies itself to something, and especially to itself: an "inert" object f cannot act on something, but can only be acted. So, *it is the process F which applies itself to something.* Therefore, we may consider self-application

either as the application of the activity F to its characterization, the graph f *or* as the application of the activity F to the activity **F**.

The *latter* application is not *a priori* absurd, as some activities like "go from a thing to this thing itself" or "go from a thing to a fixed thing a", where things may be activities, apply themselves to any activity, and especially to themselves (they can be considered as interpretations of I and Ka, where I and K are the identificator and the cancellator of combinatory logic). Neither is the *former*, which is not a "*self*-application" in the litteral sense of the word "self" (in theory of algorithms usually we apply a partial recursive function F to its notation, e.g., to its Gödel number f). This "self"-application is only forbidden for the activity prescribed by (Γ), by one of the (non evident) axioms of Set Theory, say the axiom of

4. The graph is such that there is at most a second element of a point of the graph. F is undefined for x if the search does not succeed.

foundation which makes it unfulfillable when the object x of (Γ) is the graph f of F.

2.2 The notion of function *as process* does not ask for having previously the notion of the objects (arguments) to which it applies itself and of the objects (values) to which it possibly leads, opposed to the set theoretic notion of function *as graph* (the notion of graph presupposes the notion of the ordered pairs of which it is the set). A simple instance of function as process is the identificator I which, when applied to an arbitrary object, leads to this object itself. This identificator, as a process, is clear "per se". If we take for "objects" the sets, I cannot be a set, hence cannot be a graph (if we extend the notion of graph by taking classes of ordered pairs for graphs, I may be a class in this extended sense, but if we take for "objects" the classes, again I cannot be a graph in this extended sense of objects). We have in Set Theory only "approximations" of I: for instance, if x is a set and I_x is such that for every $y \epsilon x$ we have $\mathsf{I}_x y = y$, then I_x is such an approximation of I. We have a similar situation for $\mathsf{K}a$.

These remarks throw some light upon the notion of application to which we return in order to compare the point of view of Von Neumann and the one of the combinatory logicians (which is our own). In the first place, the "active" one, of its operation of application [,], Von Neumann puts a notation, say f, for a function F (thought as a process), and in the second place, the "passive" one, he puts a notation for an argument (possibly a function, i.e., a I-II object). So, in the expression $[f, x]$ (which has the same meaning as Fx), f designates the *acting* object (more exactly *process*) F, and x the *acted* one.

Combinatory logicians get their own notion of application by dropping the notion of process from the notation f, which now becomes a notation for an inert object, this object characterizing the process F in the above case,[5] but being, in the general case, completely arbitrary, and letting the notion of process to lie now *entirely* in the operation [,] itself: this is this operation (thought as a process) which acts on two arbitrary objects x, y (thought as inert objects) and products possibly the object $[x, y]$.[6]

3 *Partial operations and extension of languages by creation of abstract objects* At the very beginning of mathematics, we use *operations* for producing terms describing the objects under consideration and *equalities* between terms as elementary relations between these objects: terms and equalities between them form the technical formal part $\mathcal{L}$ of the natural language used heavily at this stage for speaking of our objects.

As a very simple example, let $\mathcal{L}$ be the formal part of the arithmetic of positive integers $1, 2, 3, \ldots, n, \ldots$, before the introduction of the rational (positive) numbers. The alphabet of $\mathcal{L}$ contains symbols (or "elementary" terms) for designating positive integers, a symbol . for

5. Think of graphs or Gödel numbers.

6. The *process* F is $[f, \ ]$ where f is, for instance, a graph, a Gödel number, . . .

designating the fundamental operation of multiplication, parentheses for preventing collision of terms, = for the relation of equality. Terms of $\mathcal{L}$, describing positive integers, are $\alpha . \beta$ where α and β are already constructed terms. Formulas of $\mathcal{L}$ are $\alpha = \beta$, where α and β are terms.

Faced to the problem of inversion of the operation ., mathematicians introduced new terms in $\mathcal{L}$ (so extended in $\mathcal{L}_1$), the *"fractions"* α/β, first as a mere auxiliary shorthand for describing the positive integer γ such that $\alpha = \beta . \gamma$, where α and β describe positive integers. At its beginnings a language is created and *used* for speaking of objects (here the positive integers), the form of its terms and relations (to which not much attention is paid) seems inseparable from their interpretations. So more general and meaningless terms such that 7/3, if considered, would be rejected at once as useless, had not the old Greeks found a new (geometrical) interpretation of fractions, available too for the primitively meaningless ones; they were considered as "operators of measurement" and named, with this interpretation, *"rational numbers"*. Such a *geometrical* interpretation was not so immediate when the problem of inversion of addition induced to consider meaningless expressions such as 3-7, which perplexed for a moment medieval mathematicians as discloses to us the strange qualifyings ("deaf", "false") they gave to our "negative" numbers. Worse was the situation for "imaginary numbers", used with reluctance three centuries before a geometrical interpretation was found for them, for finding ordinary solutions of third degree equations, through "absurd" equalities containing such "absurd" expressions.

Returning to the case of rational numbers let us give to the fractions a *"syntactic"* interpretation (without any geometrical consideration) as rational numbers.

First we extend $\mathcal{L}_1$ to $\mathcal{L}_2$, simply by removing all the restrictions set on $\mathcal{L}_1$-terms so that they describe integers: so $\mathcal{L}_2$ contains meaningless terms such as 7/3. Second we will give a meaning to these meaningless terms by extending $\mathcal{L}_2$ to $\mathcal{L}'$ through an adjunction to $\mathcal{L}_2$ of a formal part of its metalanguage, $\mathcal{M}\mathcal{L}_2$, in such a way that $\mathcal{L}'$ become able *to speak of* $\mathcal{L}_2$-terms, and not only to *use* (some of) *them as tools* for speaking of positive integers. It will be expedient to introduce in $\mathcal{L}'$ the $\mathcal{M}\mathcal{L}_2$-formula $\mathrm{O}(\alpha)$ meaning "the $\mathcal{L}_2$-term α describes a positive integer" with its negation $\overline{\mathrm{O}(\alpha)}$. Next we introduce in $\mathcal{L}'$ the relation "=", between $\mathcal{L}_2$-terms, defined by induction on the number of / as follows:

$$\alpha/\beta = \alpha'/\beta' =_{Df} \alpha . \beta' = \alpha' . \beta$$

It is easy to prove that this relation is a relation of equivalence, and even a congruence with respect to . and /. Then we define the "positive rational numbers" as the classes of equivalence of this relation. So a fraction, like 7/3, describes the class of equivalence in which it lies i.e., the corresponding "positive rational number". We could proceed the same way for defining algebraic integers (here $\mathrm{O}(\alpha)$ would mean "the term α describes a natural number") or for defining complex numbers ($\mathrm{O}(\alpha)$ would mean "the term α describes a real number").

The partial function (from integers to integers) expressed by / in $\mathcal{L}_2$ is a very simple example of partial recursive functions, one of the most important objects of the Theory of Algorithms. In this theory we are frequently induced to contemplate meaningless terms such as fn, where n is an integer and f a partial unary recursive function, undefined for the integer n. As we cannot decide effectively for every pair f, n if f is defined for n, it is not so easy as in the simple preceding case, to give in an analogous way a meaning to meaningless expressions of the type fn[7]: so they remain usually meaningless and there is some reluctance to use them (and some hesitation about how to do). About the reluctance to use meaningless expressions we will study in the next paragraph the Hilbert's programm. About the difficulties which we may encounter in trying to give them a sense, we defer to the section **6**, on diagonalization process.

4 *Hilbertian "ideals"* Hilbert was the first to shake off the old repulsion with regard to meaningless terms and to make them an important tool of his theory of foundations (Hilbert's programm). The processes of abstraction we used for introducing rational numbers are quite simple, and the new abstract objects quite clear. It may be proved easily that the new language ($\mathcal{L}'$) obtained by the introduction of these new terms is a conservative extension of the old one ($\mathcal{L}$). But, as Shanin remarked in his "Critic of classical mathematics", such "abstractions" are not the only way through which new objects enter into mathematics. They enter too by way of "idealizations", as the "idealization of actual infinite" which produces the Cantorian notion of set. Abstractions (which generally appear when we extend a supposed sound language $\mathcal{L}$ into a new one $\mathcal{L}'$, as we did in section **3**), when alone, never produced difficulties. But Frege's "principle of abstraction", which must be held for evident if we admit the unrestricted *idealization* of actual infinite, has for consequence Russell's paradox, a contradiction.

So Hilbert was confronted to a first order completely formalized language $\mathcal{L}'$ (say the language of set theory), the soundness of which was primitively based on the accepted (as evident) meaning of its terms, and this meaning itself was based on the acceptation (as evident) of the actual infinite. Due to well known contradictions, he was forced to contemplate the eventuality of a vanishing of meaning of some terms in $\mathcal{L}'$, entailing the vanishing of meaning of complex terms constructed from these one, then the vanishing of meaning of formulas constructed from those now meaningless terms, and finally the vanishing of meaning of deductions (as we name arguments in completely formalized languages) which contain such meaningless formulas. Through these vanishings, primitively meaningful terms and formulas become pure meaningless strings of symbols of $\mathcal{L}'$, and primitively "convincing" deductions of $\mathcal{L}'$ become purely formal derivations (i.e., pure sequences of strings of symbols of $\mathcal{L}'$, constructed

7. We could do so in looking at fn as describing the process of calculus of fn itself, not its result, and considering on a par the fact it terminates or the fact it does not terminate.

according to some rules), devoid of any convincing power. Nevertheless, *as Hilbert wanted to keep untouched the language* $\mathcal{L}'$, he did not banish, as useless ballast, these formal entities. On the contrary, he accepted them without even trying to give them a new meaning *by themselves*, say a "local" meaning. They become only characters of a certain game, their "global" meaning consisting only (as for pawns, queen, bishop, and king in Chess) of their role in the whole game.

More exactly, Hilbert separated the terms of $\mathcal{L}'$ between *real* and *ideal* objects (for meaningful and meaningless terms), formulas of $\mathcal{L}'$ between *real* and *ideal* propositions (for meaningful and meaningless formulas), formulas), derivations of $\mathcal{L}'$ between *real* and *ideal* deductions (for convincing deductions and purely formal derivations). The *"real language"* $\mathcal{L}$ is the collection of the real propositions, which speak about real objects only, and its deductions are the real deductions of $\mathcal{L}'$. Hilbert choiced what he named "real" in such a severe way that nobody may doubt of the perfect soundness of the real language and of the convincing power of its deductions ($\mathcal{L}$ was seen later to be the Skolem-Goodstein language of primitive recursive arithmetic). And he said that we will be compelled to accept $\mathcal{L}'$ as a *correct* and *useful* language after having given a rule replacing effectively each (real or ideal) deduction in $\mathcal{L}'$, terminating at a real proposition, by a real deduction (in $\mathcal{L}$) with the same end-formula. The impossibility to deduce in $\mathcal{L}'$ any real proposition, without having simultaneously a real deduction of it in $\mathcal{L}$, secures the *correctness* of $\mathcal{L}'$. Its *usefulness* comes from the fact that ideal deductions are generally shorter than real ones. He laid down the programm to find such a rule, programm which was not fulfillable, as Gödel showed us, due to the excessive severity of the choice of "real" entities.

Remark 1: Hilbert's point of view is, in some way, opposite to the point of view of section **3**: the latter is the one of mathematicians of "constructive" tendancy, which try to progress by larger and larger *extensions* (as the extension from $\mathcal{L}$ to $\mathcal{L}'$ in section **3**). The former is the one of mathematicians of "speculative" tendancy, which describe in a language $\mathcal{L}'$ "ontological" concepts analogous to those of physicists (e.g., the "idealization of actual infinite" in the language $\mathcal{L}'$ of set theory): even most of these mathematicians are not busy with foundations problems and look at their—possibly inconsistent—$\mathcal{L}'$ as a process of trial and error analogous (yet more abstract) to the theories of physicists. For them, an inconsistency in $\mathcal{L}'$ would be an inadequacy and would appeal to a correction of $\mathcal{L}'$ exactly as the eventual inadequacies of a physical theory appeal to an eventual correction of it. Hilbert thinks that an already given (no matter its origin) *mathematical* theory $\mathcal{L}'$, *qua mathematical*, must be submitted to a preliminary test, say must be proved to be *"reductible"* to a sound language $\mathcal{L}$ (by proving that $\mathcal{L}'$ is a conservative *extension* of $\mathcal{L}$).

Remark 2: Hilbert showed us that we need not to give a meaning to every term: we may perfectly, as he did, cope with meaningless terms, and hence with meaningless formulas and deductions, and *let them remain in* *this state*. When we take the resolve to let them so, we will use the Hilbert's word "ideal" instead of "meaningless": meaninglessness will be considered as a *provisional* state, idealness as a *definitive* one.[8] We will use from now on (except otherwise specified) $\mathrm{O}(\alpha)$ and its negative $\overline{\mathrm{O}(\alpha)}$ for, respectively, "the term α is a real object", "the term α is an ideal object".

Remark 3: The failure of Hilbert's programm shows us that it is dangerous to divide *a priori* the terms between real and ideal objects. So we will do this separation *a posteriori*, by taking O as a *primitive* of our system, to be defined (and with it the notion of real and ideal) only on the basis of *the whole* system.

5 *Objective, syntactic, and idealized languages* In section **3**, meaningless terms such as 7/3 were banished from $\mathcal{L}_1$, present but not *used*[9] in $\mathcal{L}_2$, present, and after being given a meaning, *used*[9] in $\mathcal{L}'$. We will name "*objective*" such languages as $\mathcal{L}$, $\mathcal{L}_1$, $\mathcal{L}_2$, $\mathcal{L}'$, where every used[9] term describes an object (is "meaningful").[10] In the sequel, we limit ourselves to languages where terms are constructed from the elementary ones through the operation of application [,], and formulas are constructed from equalities[11] as elementary formulas.

For objective languages, we are induced naturally to accept the

Frege-Church principle *If a subterm α or β of a term $[\alpha, \beta]$ is meaningless,*[12] *it is the case too for the whole term $[\alpha, \beta]$.*

This principle may be *extended* to *formulas* by considering as meaningless an equality between terms not both meaningful, and as meaningless a formula, a subformula of which is meaningless (for formulas we may use the term "fictive" instead of "meaningless"). We may extend it again to the sequences of formulas which are deductions, by considering them as "meaningless" or "fictive", when one of their members—a formula—is fictive. In an objective language, after having recognized that a term is meaningless (which is of interest and sometimes not easy), this term is no more used[13] and is let aside (or it could be erased as well).

---

8. "Meaningless" is a simple lack of information, "idealness" is the decision to do not bring information.

9. As tools of discourse.

10. Clearly it is in order to get an objective language that Von Neumann separates his objects between I- and II-objects and adopts his conventions for [,].

11. We consider here "declarative" equality, putting aside for the time being "definitional" equality $\alpha =_{Df} \beta$ which can be interpreted as a command "replace the definiendum α by the definiens β".

12. I.e., describes no object.

13. As tool(s) of discourse.

The equality $\alpha = \beta$ (*"objective"* equality) of an objective language is the following relation between *objects* (defined simultaneously with its negation $\alpha \neq \beta$):

$\alpha = \beta$ means: the object described by α *is the same as* (or: *is undiscernible from*) the object described by β.
$\alpha \neq \beta$ means: the object described by α *is not the same as* (or: *is discernible from*) the object described by β.

If it is not the case that both α and β describe objects (i.e., if one of the terms α, β is meaningless), as it may happen e.g., in $\mathcal{L}_2$ of section **3**, then under this interpretation $\alpha = \beta$ and $\alpha \neq \beta$ are meaningless strings of symbols, and after being recognized such (which can be of interest), being no more used[13] may be let aside or could be erased as well. But we may look at the objective equality, *no more as a relation between the objects* described by the terms α, β, *but as a relation between these terms themselves* (as such, this relation no more belongs to our objective language, but to its syntax). In this new interpretation, we get a *partial* predicate on terms, $\alpha = \beta$, defined only when both of them describe objects and in this case having the same meaning as the *objective* equality. We get so the "*partial* equality" (Kleene's "weak" equality, IMM, 1952, p. 328).[14]

When we will formalize our languages in Predicate Logic, partial equality makes difficulties as Predicate Logic considers only *total* predicates on the universe of its objects (the common range of its variables, the so-called "Universe of Discourse") and especially a *total* predicate of equality. It is not fitted to the treatment of partial (possibly meaningless) predicates such as partial equality. We do not try to adapt logic to this new situation, but, keeping logic untouched, we must redefine equality $\alpha = \beta$ between possibly meaningless terms α, β in such a way that *it keeps always a meaning*: but, apart this last property, the equality we have in view is again, as partial equality, a "syntactic" extension of the objective equality, a relation between the *terms* of the objective language, belonging to its syntax. We set

$\alpha = \beta =_{Df}$ *either* both α and β describe objects, and these objects are just the same (are undiscernible)
or it is not the case and $\mathrm{E}(\alpha, \beta)$

where $\mathrm{E}(\alpha, \beta)$ is a relation of equivalence between *terms* α, β.

If only one of the terms α, β describe an object, then α, β are discerned on the basis of this fact: so we have "not $\mathrm{E}(\alpha, \beta)$", then $\alpha \neq \beta$. But if both α, β do not describe objects, many possibilities remain open for defining

14. We keep the name "objective language", when objective equality is looked at as partial equality.

$\mathrm{E}(\alpha,\beta)$. As the terms α, β are undiscernible on the basis of their contents (they have no contents), *if we want to discern between them*, it can be only on the basis of their form and any relation of equivalence $\mathrm{E}(\alpha,\beta)$ between the (pure) terms α, β will do the job. The simplest and strongest of these relations $\alpha \equiv \beta$ means that the terms α and β are exactly the same sequence of symbols; generally the classes of equivalence of the other $\mathrm{E}(\alpha,\beta)$ have many terms.

But to discern between meaningless terms, is, in a way, to give them a meaning, through the following process, already used in section **3**. We will name *"syntactic" equalities* the equalities $\alpha = \beta$ between terms, just defined; an extension of the primitive objective language $\mathcal{L}$ through syntactic equalities (say, a "syntactic extension" of this language) will be named a *syntactic language* $\mathcal{L}'$ *based on* $\mathcal{L}$. A syntactic language $\mathcal{L}'$ may be made objective by giving a meaning to primitively meaningless (in $\mathcal{L}$) terms, say by considering them as describing their class of equivalence. But, in the new objective language so obtained, it may subsist (provisionally) meaningless terms. If we renounce to introduce by the preceding process new objects giving a meaning to these terms, they will be "definitively" meaningless: then they may be considered as *ideal* objects, i.e., used as mere pieces of a game (with no need of interpretation). We name *"idealized"* languages those languages with ideals. As we have no more to pay attention to the differences of form of these ideal objects, we *make abstraction* of these differences, *by setting all the ideal objects to be undiscernible*: so we choose for $\mathrm{E}(\alpha,\beta)$ the weakest equivalence relation between pure terms, setting $\mathrm{E}(\alpha,\beta)$ to be the case when neither α nor β describe objects. It will be convenient to use $\mathrm{O}(\alpha)$ and $\overline{\mathrm{O}(\alpha)}$ for expressing respectively that α describes an object or that it is "definitively" not the case (i.e., that α is ideal). Using this notation we may rewrite Frege-Church principle and write the definition of the *"ideal" or "total" equality* described above (Kleene's "complete" equality, IMM, 1952, p. 328).

Frege-Church principle *If* $\overline{\mathrm{O}(\alpha)}$ *or* $\overline{\mathrm{O}(\beta)}$ *then* $\overline{\mathrm{O}([\alpha,\beta])}$. (I)

Ideal equality (total equality) $\alpha = \beta =_{Df}$ *either* $\mathrm{O}(\alpha)$ and $\mathrm{O}(\beta)$ and α and β describe the same object
or $\overline{\mathrm{O}(\alpha)}$ and $\overline{\mathrm{O}(\beta)}$ (II)

From now on we will use the symbol = for the ideal equality (and $\underset{\text{weak}}{=}$ for weak equality, except it is otherwise specified in the context).

Remark 1: The formula

$$\mathrm{O}(\alpha) \ \& \ \mathrm{O}(\beta) \rightarrow \alpha = \beta \tag{1}$$

(where & and $\rightarrow$ are the usual conjunction and conditional) is always meaningful and says the same thing as

$$\alpha \underset{\text{weak}}{=} \beta \tag{2}$$

when (2) is meaningful. When (2) disappears *as meaningless*, (1) disappears *as trivially true*: we say that (1) *"simulates"* (2).

Remark 2: We may extend an idealized language to an objective one, by considering that all the "definitively" meaningless α designate the class of equivalence of the ideal equality formed by these α: so we give *globally* a *"singular"* meaning to these α (here lies the motivation of the quotation marks surrounding the word "definitively"). This class is *the "ideal object"* (or *the "undefined"*). This singular object plays a very special role (specified by (I) and (II)) with respect to the other "regular" objects.

Remark 3: Some variances between combinatory logicians (e.g., Church and Curry) may be explained as the result of *different linguistic conventions*, on the basis of our syntactic and idealized languages. Let us, for instance, consider $\mathsf{K}a$ where K is the cancellator and a is a combinator in normal form. We may consider that the combinators which have a normal form are the *objects* of an objective language $\mathcal{L}$, the other combinators being *terms* of its syntax. So $\mathsf{K}a$ is a "syntactic" function, as it gives, for objects α of a syntactic extension $\mathcal{L}'$ of $\mathcal{L}$ (not necessarily objects of $\mathcal{L}$) as arguments, an object a of $\mathcal{L}$ as value, by $\mathsf{K}a\alpha = a$, which is true even if α has no normal form. So Curry language of pure combinatory logic may be considered as a syntactic language $\mathcal{L}'$ (based on $\mathcal{L}$). Church assimilates ideal objects and terms without normal form[15]: so, he cannot accept without caution an object like K, because if α has no normal form, the left member of $\mathsf{K}a\alpha = a$ is an ideal object by Frege-Church principle and the right one is the real object a; this is impossible, as Church accepts the rule of our ideal equality by which an ideal object is distinct from a real one. Church uses, instead of K, combinators I, J and the combinatory logic CL-I using these combinators do not give rise to such difficulties.[16] For analogous reasons, we take, instead of K, the combinator k such that $\mathsf{k}a\alpha$ is $\mathsf{K}a\alpha$ for α real and is α for α ideal.

6 *Diagonalization* When a language is sufficiently strong for expressing diagonalization, an *uncautious* assimilation of ideal object to ordinary one, without due attention paid to its very special role, may produce inconsistency.

Let F be a family of partial functions on F (i.e., their value, for an argument taken from F, may be undefined). Lower case Latin letters denote the elements of F, the eventual result of the application of a partial function $\mathfrak{f}$ (not necessarily in F) to its argument x is denoted by mere juxtaposition as $\mathfrak{f}x$. Let us define, from F, a partial function $\mathfrak{f}$, named a

15. For us this assimilation gives only one, among other possible, meaning of idealness $\overline{\mathrm{O}}$, a primitive of our system.

16. Bahrendregt ("Some extensional term models for combinatory logics and λ-calculi") proves that it is possible to equate in CL-I without introducing inconsistencies, terms without normal form.

"*diagonal for F*", by the following conditional command (1)[17]:

$$\mathrm{O}(xx) \twoheadrightarrow \text{ take } \mathrm{O}(\mathfrak{f}x) \text{ and } \mathfrak{f}x \neq xx \tag{1}$$

As usual, when the condition $\mathrm{O}(xx)$ of the command is not fulfilled, i.e., when $\overline{\mathrm{O}(xx)}$, nothing is said and everything compatible with "$\mathfrak{f}$ is a partial function on F" is permitted. Clearly *every F has diagonals* $\mathfrak{f}$. Let us now set the hypothesis

$$\mathfrak{f} \textit{ is an element } f \textit{ of } F \tag{H}$$

(H) means that F has a "inner" diagonal f. As all the f of F are supposed given before the definition of $\mathfrak{f}$, it is possible that (1), which becomes under (H):

$$\mathrm{O}(xx) \twoheadrightarrow \textit{ take } \mathrm{O}(fx) \textit{ and } fx = xx \tag{2}$$

is no more fulfillable. Taking f for x in (2), we get

$$\mathrm{O}(ff) \twoheadrightarrow \textit{ take } \mathrm{O}(ff) \textit{ and } ff \neq ff \tag{3}$$

then, by simplification

$$\mathrm{O}(ff) \twoheadrightarrow \textit{ take } ff \neq ff \tag{4}$$

Clearly (4) is not fulfillable, except if its condition fails, i.e., if $\overline{\mathrm{O}(ff)}$: so we get a *necessary* condition for the fulfillability of (2), the

Diagonalization theorem *For F having an inner diagonal f, it is necessary that* $\overline{\mathrm{O}(ff)}$.

If F is a family of total functions on F, for every $f \in F$, we have $\mathrm{O}(ff)$. So

Corollary *Every diagonal of a family of total functions on F is outside F; we say that F has only "outer diagonals".*

This gives us a general construction of a function $\mathfrak{f}$, total on F, which lies *surely* outside a family F of total functions on F. If, for every $f \in F$, we have $\overline{\mathrm{O}(ff)}$ (and especially if F is a trivial family of nowhere defined functions), every function of F is an inner diagonal for F.

The *necessary* condition furnished by the diagonalization theorem is not *sufficient* for F having an inner diagonal: we have families F such that their elements f, such that $\overline{\mathrm{O}(ff)}$, are not, nevertheless, inner diagonals of F.[18] The preceding language of "conditional commands", we used in order

17. We use, as abbreviations, the predicate O on terms α, such as xy, defined by:

$$\mathrm{O}(xy) =_{Df} \text{the term } xy \text{ is defined}$$

(i.e. describes some object, the result of the application of the function x to the argument y), its negation $\overline{\mathrm{O}}$, and $\twoheadrightarrow$ for symbol of (conditional or not) command, instead of the overworked Post's simple arrow.

18. Let, for instance, F' be a family of *total* functions on F' and let f be a function, not in F', such that $\overline{\mathrm{O}(ff)}$, but otherwise choiced arbitrarily on F'. Let, for $x', y' \in F'$, x, y denote extensions of the functions x', y' to $F = F' \cup \{f\}$, such that: $xy =_{Df} x'y'$, xf arbitrarily *de-*

to define $\mathfrak{f}$, permitted us to use equality only between *objects* (*objective* equality). We may use to the *partial* equality (written = in the present contest) between (possibly meaningless) *terms*, writing so (2) under the *unconditional* form:

$$take\ \bigcirc(fx)\ and\ fx \neq xx \tag{2'}$$

If $\bigcirc(xx)$, (2′) means just the same as (2). If $\overline{\bigcirc(xx)}$, (2′) is meaningless, by Frege-Church principle extended to formulas, as it is the case for $fx \neq xx$: so (2′) may be erased and disappears. With partial equality, the diagonalization argument runs like that:

$$take\ \bigcirc(fx)\ and\ fx \neq xx \tag{2'}$$

and substituting f for x in (2′), if the result is meaningful,

$$take\ \bigcirc(ff)\ and\ ff \neq ff \tag{3'}$$

Clearly, the last command is unfulfillable if $\bigcirc(ff)$; it *disappears* with the whole argument (2′), (3′) if $\overline{\bigcirc(ff)}$, as $ff \neq ff$ is meaningless, hence too (3′), (by extension of Frege-Church's principle to arguments). So, if the command (2′) is (vacuously for $x = f$) fulfillable, we have $\overline{\bigcirc(ff)}$.

Let us now use total equality for rewriting our first diagonalization argument. As this equality is a usual (total) predicate on terms, the predicate calculus is now available, and after replacing the command symbol $\twoheadrightarrow$ by the conditional $\rightarrow$, in order to transform commands in declarative sentences of predicate logic, we get at once the following formalization in first order predicate calculus of our first diagonalization argument:

$$(\forall x)(\bigcirc(xx) \rightarrow \bigcirc(fx)\ \&\ (fx \neq xx)) \tag{2''}$$

$$\bigcirc(ff) \rightarrow \bigcirc(ff)\ \&\ (ff \neq ff) \tag{3''}$$

by specializing x to f, then

$$\bigcirc(ff) \rightarrow ff \neq ff \tag{4''}$$

by simplification, then

$$\overline{\bigcirc(ff)} \tag{5''}$$

as $ff \neq ff$ is false.

Let us remark that the cancelling of equalities, *as meaningless* in derivation (2′), (3′) above, or *under the effect of an unfulfilled condition* in derivation (2), (3), (4) above, or *under the effect of a false premise* in the derivation (2″), (3″), (4″), (5″) above, is *essential*. If we *replace* in (2′), (3′) the partial equality by the total one, or if we *suppress* the condition in (2), (3), (4), or if we *suppress* the premise in (2″), (3″), (4″), (5″), we get a contradiction.

---

fined, $fx =_{Df} fx'$. So, the only *possible* inner diagonal for F would be f (as $\bigcirc(xx)$). But, if using our liberty of choice for f, we took, for some $g' \in F'$, $fg' = g'g'$, then f cannot even be an inner diagonal for F, as we have not $fg' \neq g'g'$.

But on what basis could be made this replacement, or these suppressions? Clearly, on the basis of a complete assimilation of the ideal object to the ordinary ones, an assimilation evidently motivated by the habits contracted and deeply ingrained since the long and remote period where languages, as the practical activities they were, were purely objective. Hence comes the tendancy to "objectivation"—or, as we say, using the Latin word "res", to "reification"—of every new object, even quite abstract, which comes under our attention, i.e., the tendancy to treat it as a material object, following this tendancy we may argue like that: the ideal object is, after all, an object "like the others". So, there is no reason to treat it apart. Then admitting implicitly that every term has a meaning (describes an object) i.e , admitting implicitly $O(fx)$ for every f, x—*every function is total*—and especially $O(xx)$, we are induced to set at once, using the total equality, $fx \neq xx$, as definition of a diagonal f for the family F of functions x (all the functions are total). More: if we use the Cantor's process of reification consisting, *first*, of an assimilation of (total) functions to their graphs, *second*, of the consideration of all the sets (hence of all the graphs) as "present" in the Plato-Cantor's Universe (resembling to the material Universe), then we may consider that F is the family of *all* the (total) functions. So f is in F, hence f is an inner diagonal of the family of total functions F: Hence a contradiction known as "Russell's paradox". So, in order to escape from this contradiction, we are induced to make the ideal object play its very special, *linguistic*, role: through the method of simulation (to be described in section **7**) its presence in a term makes this term ideal, the presence of an ideal term in a formula makes this formula "virtual", idealness and virtualness *simulating* meaninglessness of terms and formulas which are actually meaningful.

7 *Reification and simulation* Intuitively, we may conceive an "*object*" as "what of which we speak in language". But we cannot content ourselves with the vague, empirically (in a way partially impossible to foresee) extensible, notion of language as "colloquial activity", and we must precise that an object is "what of which we speak in *a* language". For some families of languages, not necessarily formal, this definition may be made utterly precise: it is for instance the case for a (finite) *text*, completely understood: this asks only for finitely many "acts of understanding", i.e., finitely many (hence performable) mental experiences[19]. Even for a completely formalized language, or a text of it, if we want to apply it (for saying something), and not only to consider it as a formal game (the whole language) or a play in it (a text in this language), the preceding *understanding* is necessary as well. This understanding lies at the origin of every objective language $\mathcal{L}$, created for being applied, i.e., for talking about some objects considered as preexisting to (hence independent of) $\mathcal{L}$, and about

19. Among the finitely many statements of a *mathematical* text, there may be "schematic statements" (Lorenzen), using free variables (as the statements of Skolem-Goodstein arithmetic) the understanding of which entails the understanding of infinitely many statements (of the schema).

which we could talk in other languages than $\mathcal{L}$. But, by going from "*language*" to "*a special* language" $\mathcal{L}$, perhaps we restrict our possibilities of expression, and perhaps we cannot say in $\mathcal{L}$ something that we could say in some extension of $\mathcal{L}$.[20] So, we are induced to take the language $\mathcal{L}$ we choose *as extended as possible*, especially by taking its universe of objects sufficiently large for securing a kind of "closure": the "*tools*" of expression of $\mathcal{L}$ must be, "*as far as possible*", *objects* of $\mathcal{L}$. We say that these tools are "*reified*" as objects.

In first order languages, the main tools of expression are, in addition to the objects, the ordinary functions for constructing terms, and, for constructing propositions, the predicates which are total functions on the universe of objects, with propositions as values. In our own first order languages, following Von Neumann, we take = as the sole predicate, and we replace the ordinary predicates by their corresponding Kleene's "logical functions" (Frege's predicates) which we note by italic letters, in such a way that if P is a predicate, $\mathfrak{p}$ its corresponding logical function, we have

$$\mathfrak{p}(x_1, \ldots, x_n) = v \leftrightarrow P(x_1, \ldots, x_n), \tag{1}$$

where $\leftrightarrow$ is the biconditional, v one of the two Frege's marks (truth values) v, f (v, the "truth", f, the "falsity") and

$$\mathfrak{p}(x_1, \ldots, x_n) = \mathsf{f} \leftrightarrow \neg(\mathfrak{p}(x_1, \ldots, x_n) = \mathsf{v}) \tag{2}$$

where $\neg$ is the negation and (1) and (2) are understood as definitions of their first member. Every formula $\mathcal{P}(x_1, \ldots, x_n)$, with $n \geqslant 0$, of a first order language defines a predicate P, hence the corresponding logical function $\mathfrak{p}$. Logical functions may be considered as special cases of functions. For the first order language $\mathcal{L}'$, to be described below, some functions are *completely* reifiable, i.e., for such a function $\mathfrak{g}$ there is an object g of $\mathcal{L}'$ (a "function") such that, for every objects x_i of $\mathcal{L}'$, we have $g(x_1, \ldots, x_n) = \mathfrak{g}(x_1, \ldots, x_n)$: it is the case for the functions $\mathfrak{g}$ *explicitly defined* from objects of $\mathcal{L}'$ ($\mathcal{L}'$ is "closed under explicit definition"). But for logical functions in general, we must interpret "reification *as far as possible*" as "*partial* reification", i.e., we may write

$$p(x_1, \ldots, x_n) = \mathfrak{p}(x_1, \ldots, x_n),^{21} \tag{I}$$

where p is an object (a "function") of $\mathcal{L}'$, only for the objects $x_1, \ldots, x_n$ of $\mathcal{L}'$ such that the formula of $\mathcal{L}'$, $\mathcal{P}(x_1, \ldots, x_n)$, of which $\mathfrak{p}$ is the corresponding (as above) logical function, has the property of "*factualness*",

20. Turing (Ordinal Logics) and, following him, Feferman (progressions of theories) created something which resembles the informal (and vague) notion of language, as creative activity (especially, creative of languages more and more extended). But if such a progression is looked at as a whole, it may be subsumed into a unique language, and, if not, it calls to an intuitive notion of extension of languages by using metalanguages, formalizing them, and so on, which has difficulties of its own.

21. And then the equivalence in $\mathcal{L}'$: $p(x_1, \ldots, x_n) = \mathsf{v} \longleftrightarrow \mathcal{P}(x_1, \ldots, x_n)$.

a property "*simulating*" meaningfulness. Given a $\mathcal{L}'$-formula $\mathcal{Y}(x_1, \ldots, x_n, y)$, we may find an object g of $\mathcal{L}'$ (a "function"), such that, for the $x_1, \ldots, x_n, y$ for which this formula is factual and functional with respect to y, we have in $\mathcal{L}'$

$$\text{(II)} \qquad y = g(x_1, \ldots, x_n) \leftrightarrow \mathcal{Y}(x_1, \ldots, x_n, y).$$

These latter results, among the most important of our theory, permit the partial reification of many functions.[22]

The construction of our language $\mathcal{L}'$ rests (intuitively) upon the "*process of simulation*". Starting from an objective language $\mathcal{L}$, we extend it to an idealized language $\mathcal{L}'$. In $\mathcal{L}'$, made objective by the introduction of the ideal object as in Remark 2 of section **5**, every term may be considered as describing an object, but some terms describe the ideal object. *We want that the idealized language $\mathcal{L}'$ simulates $\mathcal{L}$, i.e., says exactly the same things as $\mathcal{L}$ about just the real objects of $\mathcal{L}'$ (i.e., the objects of $\mathcal{L}$), and we want that a formula of $\mathcal{L}'$ which talks about objects among which is the ideal one, which would disappear as fictive in $\mathcal{L}$, disappear too (as trivially true) in $\mathcal{L}'$.*

Diagonalization (section **6**) gave us a simple example of simulation: the simulation of the partial equality (of, say, a language $\mathcal{L}$) by the total equality (of, say, a language $\mathcal{L}'$); the simultaneous disappearances of equalities in $\mathcal{L}$ and $\mathcal{L}'$ were secured by introducing in $\mathcal{L}'$, before these equalities, a premise which became false when they must disappear (as meaningless) in $\mathcal{L}$. So we reify only *partially* the diagonal $\mathfrak{f}$ (of the family F) as f, which is a restriction of $\mathfrak{f}$ to the x for which our preceding premise is true, i.e., the corresponding equality of $\mathcal{L}$ is meaningful. More generally, for the language $\mathcal{L}'$ of our theory of objects, we replace a formula Π which would disappear when meaningless in its corresponding language $\mathcal{L}$, by a formula $\mathfrak{c}(\Pi) \rightarrow \Pi$ of $\mathcal{L}'$, where $\mathfrak{c}(\Pi)$ means that the formula Π is factual (or, as we say too, "concrete"). When Π has bound variables, ranging on the whole universe of $\mathcal{L}'$, we define $\mathfrak{c}$ ("factualness") for Π by assimilating universal and existential quantifiers respectively to infinite conjunctions and disjunctions, and applying the Frege-Church principle, where meaninglessness is replaced by idealness.

A formula of $\mathcal{L}$ is meaningful only when its variables range on universes such that every term of the formula is meaningful. Let for instance x' be such a variable, the range of which is defined by the formula $\mathsf{E}(x)$ of $\mathcal{L}'$, x ranging on the whole universe of $\mathcal{L}'$ (the unary predicate E is the "sort" or "kind" of the variable x). Let $\mathcal{F}(x')$ be a formula of $\mathcal{L}$. As in $\mathcal{L}'$ all terms are meaningful, we may write in $\mathcal{L}'$ the usual definitions

$$(\forall x')\mathcal{F}(x') \leftrightarrow (\forall x)(\mathsf{E}(x) \rightarrow \mathcal{F}(x)) \tag{1}$$
$$(\exists x')\mathcal{F}(x') \leftrightarrow (\exists x)(\mathsf{E}(x) \;\&\; \mathcal{F}(x)) \tag{2}$$

22. We may add that, for the $x_1, \ldots, x_n, y$ such that respectively the formulas $\mathcal{P}(\ldots)$ or $\mathcal{Y}(\ldots)$ are virtual, $p(x_1, \ldots, x_n)$ and $g(x_1, \ldots, x_n)$ are ideal.

which permit us to do without the auxiliary (in $\mathcal{L}'$) variable x'. These formulas do not hold good in $\mathcal{L}$, as the right members may be meaningless when the left ones are meaningful; when meaningless in $\mathcal{L}$ the right members will be virtual in $\mathcal{L}'$ (so (1) and (2) introduce naturally in $\mathcal{L}'$ formulas with bound variables, possibly virtual). In order to keep track in $\mathcal{L}'$ of the $\mathcal{L}$-origin of formulas like the right members of (1) and (2), we introduce the notion of *relative* factualness (and virtualness). We say that a formula of $\mathcal{L}'$ is factual (or "concrete") *relatively to an auxiliary variable* x' (or *relatively to the kind* E *of this variable*, or *relatively to the objects of this kind*, i.e., the objects x for which $\mathsf{E}(x)$ is true in $\mathcal{L}'$) if we may get this formula as the right member of one of the definitions (1), (2), and if the corresponding left member is meaningful in $\mathcal{L}$; if this left member is meaningless, we replace "factual" by "virtual". Using relative factualness, we set new principles of partial reification (the most important calls to "factualness relatively to the classes" and lies at the basis of our "Theory of classes").

We set that two *logically* equivalent formulas of $\mathcal{L}'$ are simultaneously factual or virtual (possibly with respect to the same auxiliary variables). There are formulas which are equivalent in $\mathcal{L}'$ (for instance both true or both false in $\mathcal{L}'$), but not logically equivalent, one of them being factual, the other virtual (possibly with respect to the same auxiliary variables). This shows the "*linguistic*" character of the properties of factualness: they are properties of the *form* of a formula x of $\mathcal{L}'$, not only of what says this formula. This is the price to pay for the introduction of idealization.[23]

We restrict ourselves, for the time being, to the above intuitive explanation of our process of simulation, which will be described precisely but implicitly in the following formalization of our theory of objects, $\mathcal{L}'$. As a matter of fact in this formalization we refer to the above explanation for the understanding of our motivation, but we do not construct explicitly $\mathcal{L}$ *a priori*. On the contrary it is $\mathcal{L}'$ that we construct *a priori* and $\mathcal{L}$ could be extracted *a posteriori* from $\mathcal{L}'$, as the language formed from the factual formulas of $\mathcal{L}'$ (compare with our analysis of Hilbert's programm, section **4**). The process of simulation seems to us the best way for making the ideal object play its very special role in a first order language $\mathcal{L}'$.[24]

Remark: The reification problem, as relative to the manner of treating abstract objects, is an old one which appeared in ancient philosophy ("reification of universals") long before mathematicians, who treated their abstract objects the same way the practical language treat the material objects, using the same mode of speech (especially the usual logical particles of the empirical logic) began to question themselves on the

23. The same is true in e.g. Quine-Curry's stratification theories, where "stratification" is taken as "factualness". We have formulas which are equivalent in such a theory (for instance both true or both false), one of them being stratified and the other not stratified.

24. Which could be looked at as a kind of "conservative extension of $\mathcal{L}$" obtained through this process.

legitimacy of their attitude, especially when foundational critics began to spread. This problem is formulated and solved in different ways in modern mathematics. In Gödel's set theory it could be formulated as "When a class is it a set?". In cumulative type theory of predicates of Hilbert-Ackermann ("Grundzüge der theorestischen Logik"), it could be formulated "When a "Prädikatenpradikate" (more generally: a Prädikate" of the n° "Stüfe") is it an object (a "Gegenstand")?". Generally the reification asked for is a complete, not a partial one.

8 *Towards formalization* Starting from the opinion that the Cantorian notion of set, *with its idealization of abstract infinite*, is really an intricate notion, nearer of the general notion of Physics (space, time, material universe) than of the notions of hard core mathematics (integers, algorithms), we placed the notion of set at the end of the construction of (by us) simpler notions, i.e., objects, classes, universes, aggregates, sets, in the order of growing complexity. In the subsequent formalization we will present, in the Chapter II of this work, an idealized language $\mathcal{L}'$ of the "*proper* theory of objects". We name the objects of $\mathcal{L}'$ "entities" or (as Curry) "obs"; we note them with lower case greek letters. The *real* objects of $\mathcal{L}'$ (those different of the ideal object of $\mathcal{L}'$, i.e., the objects of $\mathcal{L}$) are noted by lower case Latin letters and named simply "objects". We define factualness of our proper theory of objects as "factualness relative to the objects". The axioms of $\mathcal{L}'$ are borrowed from "Eine Axiomatizierung der Mengenlehre" of Von Neumann: more exactly, we adopt his preliminary, combinatory,[25] logical[26] axioms, with the modifications necessitated by our point of view (section **1**). Corresponding to Von Neumann's theorems on normal forms, we have our fundamental theorems of combinatory normalization (translating the closure of our theory under explicit definition, see section **7**) and our two fundamental theorems of logical normalization (expressing I and II, section **7**). These theorems permit a more compact expression of our proper theory of objects. In Chapter III we examine the combinatory part of this theory, i.e., its theorems derived only from preliminary and combinatory axioms. Using combinators (s and k) we give a simple form of this combinatory part and we prove its consistency. In Chapter IV, we introduce, by using auxiliary variables ranging on more restricted domains than the domain of objects, and possibly supplementary axioms, the *general* notion of "*theory of special objects*": these theories are axiomatic extensions of the proper theory of objects. Then we introduce at once a theory of special objects, say $\mathcal{L}'_{C}$, the "*theory of classes*", and by successive axiomatic extensions, the "*theory of universes*" $\mathcal{L}'_{CU}$, the "*theory of aggregates*" $\mathcal{L}'_{CUA}$, the "*theory of sets*" $\mathcal{L}'_{CUAS}$. Note: See Andre Chauvin's PhD. Thesis, Universitê DeClermont, "Theorie des objects et theorie des ensembles."

Universitê d'Alger
Alger, Algeria

25. "Axioms of arithmetical construction" in Von Neumann's terminology.

26. "Axioms of logical construction" in Von Neumann's terminology.