# A short essay on the interplay between algebraic language theory, galois theory and class field theory: comparing physics and theory of computation

Takeo Uramoto
Institute of Mathematics for Industry, Kyushu University

**Abstract**

This paper is written as a technical report for our talk given at the RIMS workshop on quantum fields and related topics, held on 6th – 8th December 2021. In this talk we introduced our recent works [23, 24, 25, 26] in formal language theory to the community of mathematical physics, which concern some interplay between algebraic language theory, galois theory and class field theory. In this paper we discuss some conceptual contents of our recent works [23, 24, 25, 26] in more detail.

## 1 Introduction

The purpose of this paper is to discuss some conceptual aspects of our recent works [23, 24, 25, 26] on interplay between algebraic language theory, galois theory and classical class field theory, especially aiming to highlight geometric ideas behind theory of computation. Algebraic language theory on one hand is a branch of theory of computation (or formal language theory) that has been developed since the 1960s (cf. [17]); galois theory, on the other hand, belongs in the context of number theory and geometry. As this history shows, these two theories have been developed independently, but as discussed in [23, 24], algebraic language theory can now be reviewed as a monoid extension of galois theory in a certain precise (categorical) sense; more importantly, this viewpoint sheds a new light on classical class field theory in the sense we discussed in [25, 26]. In view of that some mathematical physicists are concerned with a version of *Langlands program* (i.e. today's well-recognized approach to non-abelian class field theory) in relation to its physical analogues in particular, it would not be so meaningless to introduce this recent development [23, 24, 25, 26] to the community of mathematical physics too in that our results in [23, 24, 25, 26] might provide some new viewpoint for non-abelian class field theory.

In this relation, it should be mentioned further that there have been several physical/conceptual (hence, more than pragmatic) discussions on communication between physics and theory of computation (e.g. [1, 13, 16]); in fact these are relevant to our naive motivation behind the considerations in [23, 24, 25, 26]. It is true that physics and theory of computation are quite different as scientific disciplines in several respects, say, in their objectives, methodologies and mathematical languages; however, it is also true that these disciplines have been communicating with each other at several levels: For instance, at the first level, some key ideas in theory of computation (say, computational complexity) play a key role in analysis of several statistical mechanical models [16]. Not only this, at the second level, there are more fundamental discussions to reduce our computability concept to some more basic physical principle, in relation to the *physical Church-Turing thesis*, cf. [13]: some version of this thesis argues that, roughly speaking, a function is computable in the formal sense of Church-Turing (i.e. definable by some formal Turing machines) if and only if it can be realized by some actual physical system of any form (either of classical or quantum), not limited to our familiar electric devices. Apparently, this thesis involves not only a claim on computability concept but also a strong claim relevant to principles of physics; in this way, physics and theory of computation even share a common fundamental interest.

Indeed, despite ostensible differences, physics and theory of computation are equally concerned with their respective classes of (continuous/discrete) dynamical systems, and have developed "geometric" frameworks to analyze them (namely, frameworks of symplectic manifolds or $C^*$-dynamical systems/various computational models such as Turing machines). In view of the above-mentioned fundamental intersection between these subjects, it seems essential to investigate, in mathematically systematic way, what they share implicitly or explicitly, not just looking at their differences. Indeed, in the literatures, several authors (cf. e.g. [1]) have been concerned with this issue and developed their respective fruitful frameworks, which motivated the current work too.

It is in this relation that we regard it reasonable to introduce our developments in [23, 24, 25, 26] to mathematical physicists, which indicate intrinsic connection of algebraic language theory to class field theory, hence possibly to non-abelian class field theory, which in turn has something solid to do with physical variant of Langlands correspondence as a matter of geometry. In fact, our intention is to re-locate theory of computation coherently in some geometric context as mathematical physics (or as its discrete fragment), especially in seamlessly hierarchical manner. Indeed, to our thought, this is the very subject of (relative) theory of languages, or *foundation of mathematics*, which should be able to answer to the famous question concerning "unreasonable effectiveness of mathematics" [27] in natural science (cf. §4).

In this paper, we will start with a conceptual discussion on general languages in linguistic sense, aiming to review algebraic language theory in wider perspective (§2). Although algebraic language theory itself has eventually been developed almost separately from linguistics, it indeed has its roots in the work of Chomsky in the 1960s; recalling this roots is not for historical purpose, but necessary to highlight the linguistic role/aspect of mathematics (or geometry) for physics in particular. After this discussions, we proceed to our recent works [23, 24, 25, 26] that demonstrate intrinsic connection of algebraic language theory to classical class field theory (§3). The most interesting ingredients of this results are in fact found in our technicalities[1]; but not only technicalities, we are also concerned with conceptual ingredients equally. From this standpoint, the subject of §3 is to relocate algebraic language theory in an unconventional context, i.e. that of general geo-metry, by highlighting some analogies based on [23, 24, 25, 26]; through this relocation, we then compare (mathematical language of) theory of computation with (that of) physics in §4.

**Acknowledgement** We are grateful to Izumi Ojima, Hayato Saigo, and Yoshihiro Maruyama for long-time discussions concerning languages and quantum mechanics, which heavily influenced the

[1] Therefore, for those who are not concerned with conceptual discussions, it is safer (and in fact logically possible) to read [23, 24, 25, 26] purely as arithmetic results; we separated the current consideration from the arithmetic paper [26]. To our thought, nevertheless, it seems inevitable to enter into basic linguistic (and cognitive-scientific) issues if we are concerned with common grounds shared by physics and theory of computation, cf. §4.

author's philosophy on these subjects. We are also grateful to Shigeru Taguchi, who provided us a valuable opportunity to work on phenomenology when we were a member of Nagahama Institute of Bio-Science and Technology; and to the organizer of the RIMS workshop for inviting us to give a talk there. This work was supported by the Research Institute for Mathematical Sciences, an International Joint Usage/Research Center located in Kyoto University.

# 2 Formal theory of languages

Here we recall some linguistic backgrounds of our works [23, 24, 25, 26]. These linguistic discussions are intended to highlight the fact that the studies of languages are indeed comparable to physics, and for some preparation of our arguments in the last section §4; we will come back to this linguistic discussion after some geometric comparisons (§3) of our works with the galois theory of differential equations.

## 2.1 Linguistic background

As outlined in §1, we are first concerned with considering the principal role of *languages* in general; indeed this is deeply related to our major concern in understanding the "unreasonable effectiveness of mathematics" *in a mathematical way.* Unarguably it is through languages (in a broad sense) that our subjective thoughts are externalized and informed to others; furthermore, arguably though, we might manipulate, control and build our own subjective thoughts with the aid of language. In this respect, a classical approach in linguistics was to investigate the structure of languages (or written texts) as a clue to study our mind (or linguistic ability) indirectly.

To clarify the point, this approach in linguistic can be naturally compared to traditional methods in physics, namely, measurements by specific instruments: For instance, in thermodynamics, there had been long discussions on the mechanism of sensation of heat (and cold); but it was recognized that the heats of gas are correlated to gas expansions. Since the volume of gas itself can be measured objectively (or quantitatively), one can utilize this phenomenon as symbol of heats, whatever they are; this led to invention of thermometer, with which thermodynamics could step toward an exact science. Of course, thermometers themselves do not tell us mechanisms of heat phenomena; however they provide objective clues to develop and test our theories of heat [28, 15].

The same is true for linguistic phenomena: Although it is mysterious neuro-scientifically what happens actually in our brane when we write/read some sentences, in other words, how symbols (or texts) are connected with our thoughts (or the meanings) inside brane, it is unarguable that written texts themselves are externalized and hence objectively observable; and also that the structures of texts, if grammatical, represent the writer's mind. For an apparent instance, the text that the reader is reading right now represents (approximately) what the current author has in mind; and without this texts, we would never be able to convey our personal thoughts to the reader; this is the very role of language in general. Similarly to the above relation between thermometers and heat phenomena, text data themselves do not tell us mechanisms behind linguistic phenomena; nevertheless, as what thermometers did in thermodynamics, observable text data provide us objective clues to develop and test our theories of linguistics.

Indeed, in his famous work [11, 12], Chomsky analyzed texts carefully to develop a fruitful theory concerning *generative grammars*, which theorizes mechanism of human ability to judge and create "grammatically correct" sentences: For instance (cf. [12]), for those who are familiar with English, it is not hard to see that the sentence "Colorless green ideas sleep furiously" is grammatically correct, while the sentence "Furiously sleep ideas green colorless" is not, despite both sentences are equally non-sense and consist of the same set of words; this exemplifies that the grammatical correctness is not equivalent to meaningful-ness; hence we judge, as grammatically correct, even sentences that we never use in our daily communications. He then proceeded to show, by examples, the empirical fact that we have the ability to create grammatically correct sentences from very limited experiences; from these, he argued that our ability to learn grammatically correct sentences cannot be explained by statistical leaning models but might have some more innate mechanism. Importantly, his theory satisfies several criteria for exact sciences in that it is based on careful analysis of observable text data, and by them, proposes rigorously defined mathematical models (i.e. generative grammar) for grammatical structures of languages, by which he could analyze several linguistic issues; in relation to our current work, it is also of concern that his models are in fact equivalent to *Turing machines* (cf. §2.2).

## 2.2 Formal theory of languages and computation

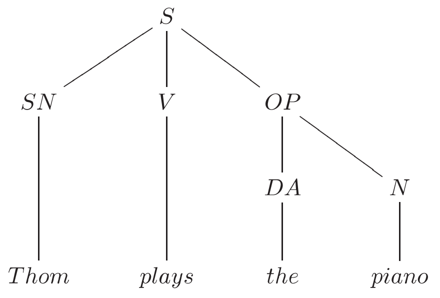
To be more formal, notice first that English sentences by themselves are sequences of alphabetical symbols, namely, capital/small letters, periods, commas and spaces; however, not just sequences but being ordered correctly, such sequences of symbols can be "grammatical" to make sense; otherwise, the sequences of symbols are "grammatically incorrect" to have no meaning. Here, the *grammar* of English is by definition a rule to prescribe the "correct" English sentences; of course, since English is a natural language, we do not know in advance any God-given rule (or grammar) of correct English sentences in the same sense as that in physics we do not know the God-given rule of our universe in advance; the theory of *generative grammars* was introduced to describe grammars of languages in a rigorous mathematical manner, and to discuss and test them by comparing with observable text data in the same way as we use mathematical models in physics.

To be specific, let $\Sigma_E := \{\mathrm{a}, \mathrm{b}, \mathrm{c}, \cdots, \mathrm{z}, \mathrm{A}, \mathrm{B}, \mathrm{C}, \cdots\}$ be the above mentioned set of alphabetical symbols for English. Then an English sentence is a priori a sequence of symbols in $\Sigma_E$; denote by $\Sigma_E^*$ the set of all finite sequences $w = a_1 a_2 \cdots a_n$ of *letters* $a_i$ in $\Sigma_E$ (also called as *finite words over the alphabet* $\Sigma_E$). Of course, as mentioned above, not all sequences of letters in $\Sigma_E$ are (grammatically) correct; hence, the set $L_E$ of all (grammatically) correct sentences is a proper subset of $\Sigma_E^*$, that is, $L_E \subset \Sigma_E^*$. A major subject of English grammar is to specify the patterns that characterize correct sentences, or more formally, patterns that determine the members of $L_E$ and distinguish them from general $w \in \Sigma_E^*$; such a set of patterns, if possible to specify, is called a *grammar* of English. The concept of generative grammar is then a mathematically formal definition of the informal term "grammar (or patterns)" here, which prescribes admissible *phrase structures* of sentences in terms of combinatorics of finite words.

The precise definition of generative grammars is not necessary for the aim of the current paper, cf. [12]; but briefly speaking, the basic idea is to abstract the traditional procedure of analyzing the phrase structures in sentences: For instance[2], we notice that the sentence "Thom plays the piano" consists of three major parts, i.e. the subjective noun (SN) "Thom", the verb (V) "plays", and the objective phrase (OP) "the piano"; furthermore, the objective phrase (OP) by itself has the inner structure consisting of the definite article (DA) "the", and the noun (N) "piano". In other words,

[2] For more nicer examples of phrase analysis, the reader is referred to some literature in linguistics, say, [12]. The point here is that we have a formally defined mathematical model (generative grammar) of the procedure of phrase structure analysis of sentences. The reader who is not concerned with linguistic models can skip to §2.3, where we discuss our major concern.

this sentence can be analyzed into the following *purse tree*, which represents an abstract structure of the sentence, and where each intermediate symbol (e.g. SN, V, etc.) represents the abstract-level grammatical (semantic) role of each phrase:




(2.1)

Looking from the root of this tree (namely the vertex S, representing "sentence"), this tree indicates the fact that this "sentence" S consists of three parts SN, V, OP; and the phrase OP consists of two parts DA, N; each of these parts is then replaced by the concrete words ("Thom", "plays", "the", "piano"). Thus, more formally, this procedure of phrase structure analysis can be seen as (inverse of) that of repeatedly rewriting finite words: S $\to$ SN V OP $\to$ SN V DA N $\to$ Thom plays the piano. The formal definition of generative grammars is a mathematical abstraction of this procedure; and rigorously, a generative grammar $\mathscr{G}$ is given as a finite set of rewriting rules over finite words such as S $\to$ SN V OP and DA $\to$ the. Given such a set of rewriting rules (i.e. a generative grammar) $\mathscr{G}$ over the alphabet $\Sigma_E$, it naturally defines $L(\mathscr{G}) \subseteq \Sigma_E^*$, namely $L(\mathscr{G})$ consists of finite words $w \in \Sigma_E^*$ which can be deduced from the root symbol S by repeatedly applying the rewriting rules of $\mathscr{G}$; we say that the generative grammar $\mathscr{G}$ *generates the language* $L(\mathscr{G}) \subseteq \Sigma_E^*$; further, for a given $L \subseteq \Sigma_E^*$ (say, $L_E$), we say that $\mathscr{G}$ is a *grammar of* $L$ if $L = L(\mathscr{G})$.

By definition, the membership $w \in L(\mathscr{G})$ means that there exists a finite-step deduction S $\to \cdots \to w$ of $w$ from S by applications of rewriting rules of $\mathscr{G}$; intuitively, this deducibility says that the sequence $w \in \Sigma_E^*$ matches the admissible pattern that the grammar $\mathscr{G}$ prescribes. In this way, a generative grammar $\mathscr{G}$ could define the language $L(\mathscr{G}) \subseteq \Sigma_E^*$ in a *deductive* way. But the phrase structure analysis, or the way in which we understand the phrase structure of a sentence, is more *inductive* in that the procedure of understanding the phrase structure of a sentence $w$ is *inverse* to the deductive procedure S $\to \cdots \to w$. Therefore, technically speaking, it is not apparent from this definition whether the membership $w \in L(\mathscr{G})$ is effectively decidable; this issue would be of natural concern from the standpoint of linguistics too, in view of the empirical fact that we can somehow detect, often immediately, the phrase structure of a sentence just from written or verbal text data. Concerning this, a fundamental result on generative grammars is that a language $L \subseteq \Sigma_E^*$ can be generated by some generative grammar in the above formal sense (i.e. $L = L(\mathscr{G})$ for some generative grammar $\mathscr{G}$) if and only if $L$ can be recognized by some *Turing machine* $\mathscr{M}$, i.e. computational model with which our computability concept is formally defined; in particular, when this is the case, a sequence $w \in \Sigma_E^*$ is a member of $L(\mathscr{G})$ if and only if the Turing machine $\mathscr{M}$ recognizes $w$ (cf. [21]); in other words, generative grammars are Turing complete. More generally the hierarchy of formal languages defined in terms of (restriction of) generative grammars (i.e. the *Chosmky hierarhcy*) is known to be in a good correspondence with hierarchy of computational models given in terms of their ability of memory.

Notable here is that, as shown in the history of physics, we can sometimes explain phenomena of concern by formal analysis of formal models[3]; this is also true for the theory of generative grammars [11]. What we can do with this formal theory is to compare formal language models $L(\mathscr{G}) \subseteq \Sigma_E^*$ with the actual language $L_E \subseteq \Sigma_E^*$ through individually observable text data $w \in L_E$[4]; indeed this can be naturally compared to the relationship in physics between mathematical models (or theories) of physical phenomena and their observable experimental data. Therefore the validity of each formal model inevitably depends on the results of comparisons to their respective target phenomena; but nevertheless, it is thanks to formally defined models that we can formulate *rigorously testable* claims on our target issue. Indeed, for instance, Chomsky [12] concluded that the set $L_E$ of correct English sentences is not a *regular language* (cf. §3.1)[5] by showing that the language $L_E \subseteq \Sigma_E^*$ does not satisfy a certain combinatorial property that every regular language must satisfy. This is a good example that formal language models and their purely mathematical analysis could deduce a linguistically interesting consequence.

## 2.3 Our major concern

In general, beyond the context of linguistics (hence, beyond natural languages such as $L_E$), formal language theory studies the structure of *arbitrary* languages, namely, arbitrary sets $L \subseteq \Sigma^*$ of finite words over arbitrary alphabet $\Sigma$ of letters; the above Turing completeness of generative grammars holds in this general setting. At a practical level, the framework of formal languages and generative grammars has been applied to the design of artificial languages (i.e. programming languages). But our major concern is in the point that, at this abstract level, the classical theory of formal languages and computation is not only about languages or computers but actually would have a broader scope of applications in nature. To highlight the point, it would be meaningful to compare our subject to physics, or more specifically, to the classical Newtonian mechanics: Based on experiments, physicists developed a framework of classical mechanics, where the *differential calculus* plays a key role in its mathematical formulation and developments as exact science; but the differential calculus itself has a broader scope of applications, apart from its original context; this is simply because the concept of real numbers and their differential calculus could have independent meaning in their own right, apart from their physical interpretations. The same is true for the theory of formal languages and computations: at the most abstract level, this discipline can essentially be about discrete dynamical systems of any form, and amongst others, best characterized by its unique concept of *computational complexity* and formal method to measure computational complexity in quantitative manner, which makes sense for any kind of discrete space/time series, thanks to the abstractness of the concept of finite words $w = a_1 a_2 \cdots a_n$.[6]

To be precise the hierarchy of computational complexity (of languages) is conventionally defined with respect to several measures of complexity and ability of formal computational models such as Turing machines, *pushdown automata* (cf. [21]), and *finite automata* (cf. §3.1). The major differences between these computational models are in the structure of the memory devices that they can use in

---

[3] This is related to our major concern on "unreasonable effectiveness of mathematics".

[4] We note that the language $L_E \subseteq \Sigma_E^*$ (of "correct" English sentences) itself is actually not well-defined unless we can define the meaning of "correct", which in nature makes sense only through observations on individual sentences.

[5] A language $L \subseteq \Sigma^*$ is *regular* if it is recognized by *finite automata*; this is equivalent to that the language $L$ has a grammar $\mathscr{G}$ of a certain restricted form, i.e. a *type-3 grammar* in the Chomsky hierarchy. Therefore Chomsky's result says that the language $L_E$ of correct English sentences is never defined by type-3 grammar in principle.

[6] This should be compared to the fact that the concept of real numbers can be used to parametrize any continuum data, thanks to their independence from any physical interpretation.

their computational procedures: For instance, Turing machines use their *Turing tapes*, where they can read/rewrite symbols and their pointer can move both to the left and the right on their tape; compared to Turing machines, pushdown automata have slightly restricted memory devices, where they can also read/rewrite symbols but they can edit their register from some restricted directions only; hence (some version of) pushdown automata can access to their past memories only from the nearest past, while Turing machines can access to their past memory at any depth if they want (cf. [21]). Then it is known that this restriction of access to past memories makes an essential difference on the ability of these computational models in the sense that the class of languages that can be recognized by pushdown automata (or equivalently, languages definable by *context-free* grammars) is a proper subset of the class of languages that can be recognized by Turing machines (equivalently, languages defined by arbitrary (i.e. type-0) grammars); some language can be recognized by Turing machine but never recognized by any pushdown automata due to the limitation of their ability to refer to the past history of computational procedure.

Having developed these characteristic results on formal languages and computational models, the theory of formal languages and computations has several notable aspects comparable to physics: The first one is that, while physics has developed mathematical understanding of our universe (or the *external reality*), this discipline tries to develop equally mathematical understanding of our mind (or the *internal reality*); indeed, back to the linguistics (cf. §2.1), formal language theory targeted at a mathematical understanding of our linguistic abilities; and the theory of computations, historically, also targeted at mathematical understanding of our computability concept. In their nature, if rightly developed, these disciplines might be *complementary* to each other (cf. §4). The second one is that, while the mathematical language for the classical mechanics, namely the differential calculus, could have their own meaning apart from their physical interpretation and be mathematically fruitful in their own right so that the differential calculus has a broader scope of applications, the mathematical language for the study of linguistics and computability, namely the theory of formal languages and computational models, have their own meaning apart from the linguistic/logical interpretations and be mathematically fruitful in their own right so that its unique idea of computational complexity can play key roles in our understandings and classifications of several (mathematical/physical) objects, say, statistical mechanical models, cf. [16]. The third one is that, furthermore, while it is unarguable that the development of physics has profound influences on our technology to understand the nature, it becomes more and more unarguable today that the development of the theory of formal languages and computations (or practically, *computer science*) has significant influence on our technology too; in fact, while physics has been recognized as the king of natural science, computer science has been growing its presence in natural science (or actually any sciences) both practically and conceptually (cf. §1; see also [1, 13]).

Of course, as mentioned in §1 too, these two disciplines are still far from each other, in particular in their mathematical languages. However, in view of the first (complementary) aspect above, as well as the fundamental intersection between these disciplines discussed in §1, it seems essential for both subjects that we are able to compare them not at a philosophical level but at some mathematically formalized level. To this end, we compare their traditional mathematical languages, namely, (linear) differential equations and finite automata. Starting from some vague analogy between them (§3.1), we formalize this analogy with some categorical terms (§3.2); based on this, we then try to highlight some common/distinct points between these two mathematical languages (§4).

# 3 Algebraic language theory, revisited

Having this program in mind, we start by reviewing the classical branch in formal language theory known as *algebraic language theory* (cf. [17]) (more specifically, *Eilenberg's variety theory* [14]). As emphasized above, formal language theory and theory of computation are best characterized by the idea of computational complexity and their hierarchical classifications. Algebraic language theory belongs in this context and is characterized by its methodology, where we use algebraic methods to classify languages; among others, the hierarchy of *regular languages* (or equivalently, the languages generated by type-3 grammars) admits most systematic classification by means of finite semigroup theory[7].

This theory on regular languages is analogous to that of linear differential equations at several levels: firstly, at an equational level (in terms of *Brzozowski derivatives* of languages), and secondly, at a cateogry-theoretic level. In fact, as discussed in [23, 24], this theory can be regarded as a monoid extension of galois theory through a certain duality theorem[8] which is a natural discrete analogue of a duality theorem[9] known in *differential galois theory* (i.e. the galois theory for linear differential equations); interestingly, this unification then sheds a new light on classical class field theory, or in particular, the theory of complex multiplication, cf. [25, 26] (§3.2).

## 3.1 The first-level analogy

As mentioned above, there are two levels of analogies between finite automata (resp. their regular languages) and systems of linear differential equations (resp. their solutions). The first-level analogy is that any regular language $L$ satisfies a certain system of equations on *Brzozowski derivatives* of languages, which represents the (minimum) deterministic finite automaton that recognizes $L$. The second-level analogy is that there is a classification theory of such "differential equations" (Eilenberg theory), which can be axiomatized in terms of the duality theorem for *semi-galois categories* [23, 24] in the same way as the galois theory of linear differential equations can be axiomatized in terms of the duality theorem for *Tannakian categories* [22, 18] (cf. §3.2).

Concerning the first-level analogy, let $L \subseteq \Sigma^*$ be an arbitrary language over an alphabet $\Sigma$; for each letter $a \in \Sigma$, we define the *Brzozowki derivative of $L$ by $a \in \Sigma$*, denoted $\partial_a L \subseteq \Sigma^*$, as follows:

$$\partial_a L \quad := \quad \{w \in \Sigma^* \mid aw \in L\}. \tag{3.1}$$

This Brzozowski derivative $L \mapsto \partial_a L$ satisfies the following Leibniz-like rules[10]:

$$\partial_a(L+R) \quad = \quad \partial_a L + \partial_a R \tag{3.2}$$
$$\partial_a(L \cdot R) \quad = \quad (\partial_a L) \cdot R + \epsilon(L) \cdot (\partial_a R), \tag{3.3}$$

where the *sum* $L + R \subseteq \Sigma^*$ denotes the union of $L$ and $R \subseteq \Sigma^*$, and the *concatenation* $L \cdot R \subseteq \Sigma^*$ denotes the language defined by $L \cdot R := \{uv \mid u \in L, v \in R\}$; also $\epsilon(L) \subseteq \Sigma^*$ is defined to be $\{\varepsilon\}$ if $\varepsilon \in L$, and $\emptyset$ otherwise.

[7] We call this fragment as Eilenberg's variety theory; but here we almost identify algebraic language theory with Eilenberg's variety theory.

[8] To be precise, a duality theorem between profinite monoids and *semi-galois categories* [23, 24], which is a natural extension of the duality theorem between profinite groups and *galois categories* (cf. §3.2).

[9] The duality theorem between algebraic group schemes and (neutral) Tannakian categories; but there are several variants [18].

[10] These are somewhat analogous to the rules satisfied by Fox derivatives on group rings.

In general, the orbit $\partial_{\Sigma^*} L := \{\partial_{a_1}\partial_{a_2}\cdots\partial_{a_n} L \mid a_1, \cdots, a_n \in \Sigma, n \geq 0\}$ of a language $L \subseteq \Sigma^*$ with respect to the actions of Brzozowski derivatives $\partial_a$ $(a \in \Sigma)$ is an infinite set; but it is well-known that a language $L \subseteq \Sigma^*$ is regular (i.e. recognized by some finite automata[11]) if and only if this orbit is a *finite* set; moreover, if this is the case, the orbit $\partial_{\Sigma^*} L$ gives rise to the minimum deterministic finite automaton (DFA) recognizing $L$. To be more specific, let $S := \partial_{\Sigma^*} L = \{L_1, \cdots, L_n\}$ be the state set (with $L_1 = L$) and define the transition function $\delta : S \times \Sigma \to S$ by $\delta(L_i, a) := \partial_a L_i \in S$. This data defines a DFA $(S, \delta)$ that recognizes the original regular language $L$ with the initial state $s_0 := L_1 = L \in S$ and the final states $F := \{R \in S \mid \varepsilon \in R\}$. By definition, the regular languages $L_i \in S$ satisfy a system of "differential equations" with respect to Brzozowski derivatives in that, for each $L_i \in S$ and $a \in \Sigma$, we have $\partial_a L_i \in S$, thus $\partial_a L_i = L_j$ for some $j$; and this relationship is equivalent to the data $(S, \delta)$, in other words, the DFA $(S, \delta)$ can be identified with this system of differential equations.

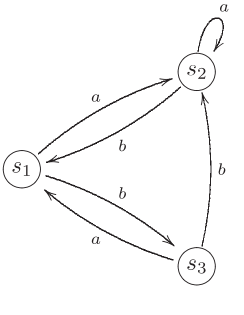
Conversely, regular languages can be seen as solutions to such "differential equations" (DFAs): Given a DFA $(S, \delta)$ over a finite alphabet $\Sigma := \{a_1, \cdots, a_m\}$, we shall *symbolically* denote this data by the following differential-equational form: let $S = \{s_1, \cdots, s_n\}$ and denote $s_{ij} := \delta(s_i, a_j) \in S$, then

$$\begin{aligned} ds_1 &= a_1 \cdot s_{11} + \cdots + a_m \cdot s_{1m} \\ &\vdots \\ ds_n &= a_1 \cdot s_{n1} + \cdots + a_m \cdot s_{nm} \end{aligned}$$

This system of equations is equivalent to the data $(S, \delta)$.

Now fix final states $F \subseteq S$, and for each $s_i \in S$, we associate the regular language $L_i$ recognized by the DFA $(S, \delta, s_i, F)$. Then these regular languages $L_i$ give the solutions to the above system of differential equations in that, if we replace $s_i$ with $L_i$ (recall that each $s_{ij}$ are some $s_k \in S$) and $ds_j$ with $dL_j := L_j \backslash \{\varepsilon\}$ (namely $dL_j$ is $L_i$ minus the "constant term" $\varepsilon$), these regular languages $L_i$ and $dL_j$ satisfy the above equations by interpreting $+$ the union of languages and $\cdot$ the concatenation. To see this, the best way is to consider some example.

**Example 3.1.1.** For instance, suppose that $\Sigma = \{a, b\}$ and the DFA $(S, \delta)$ is given as follows:




[11] For some notations and terminology on DFAs and regular languages, see the Appendix A.

then the corresponding differential equations are the following:

$$ds_1 = a \cdot s_2 + b \cdot s_3 \tag{3.4}$$
$$ds_2 = a \cdot s_2 + b \cdot s_1 \tag{3.5}$$
$$ds_3 = a \cdot s_1 + b \cdot s_2 \tag{3.6}$$

This is because, by definition, we have $s_{11} = \delta(s_1, a) = s_2$, and $s_{12} = \delta(s_1, b) = s_3$ and so on. If we take, for instance, the final state $F := \{s_3\}$, and then associate to each $s_i \in S$ the regular languages $L_i := L(S, \delta, s_i, F)$, these regular languages $L_1, L_2, L_3$ and $dL_1, dL_2, dL_3$ defined above satisfy the following equations of languages:

$$dL_1 = a \cdot L_2 + b \cdot L_3 \tag{3.7}$$
$$dL_2 = a \cdot L_2 + b \cdot L_1 \tag{3.8}$$
$$dL_3 = a \cdot L_1 + b \cdot L_2 \tag{3.9}$$

Note that these equations are equivalent to the following equations written in terms of Brzozowski derivatives:

$$\partial_a L_1 = L_2 \qquad \partial_b L_1 = L_3 \tag{3.10}$$
$$\partial_a L_2 = L_2 \qquad \partial_b L_2 = L_1 \tag{3.11}$$
$$\partial_a L_3 = L_1 \qquad \partial_b L_3 = L_2 \tag{3.12}$$

Further notice that, if languages $L_1, L_2, L_3$ satisfy this system of differential equations, then they are necessarily all regular languages by their orbit finiteness.

**Remark 3.1.2.** To highlight the analogy, we regard the equations for the DFA $(S, \delta)$ as an analogue of the following differential equations on three functions $f_1, f_2, f_3$ with two variables $x, y$ in terms of their differential 1-forms:

$$df_1 = dx \cdot f_2 + dy \cdot f_3$$
$$df_2 = dx \cdot f_2 + dy \cdot f_1$$
$$df_3 = dx \cdot f_1 + dy \cdot f_2$$

Equivalently, comparing this with the following general definition of differential 1-forms:

$$df_1 = dx \cdot \frac{\partial f_1}{\partial x} + dy \cdot \frac{\partial f_1}{\partial y}$$
$$df_2 = dx \cdot \frac{\partial f_2}{\partial x} + dy \cdot \frac{\partial f_2}{\partial y}$$
$$df_3 = dx \cdot \frac{\partial f_3}{\partial x} + dy \cdot \frac{\partial f_3}{\partial y}$$

the above equations on differential 1-forms are equivalent to the following equations similar to the language equations in terms of Brzozowski derivatives:

$$\frac{\partial f_1}{\partial x} = f_2 \qquad \frac{\partial f_1}{\partial y} = f_3 \tag{3.13}$$

$$\frac{\partial f_2}{\partial x} = f_2 \qquad \frac{\partial f_2}{\partial y} = f_1 \tag{3.14}$$

$$\frac{\partial f_3}{\partial x} = f_1 \qquad \frac{\partial f_3}{\partial y} = f_2 \tag{3.15}$$

Compare these equations of two forms with those of the form of the DFA $(S, \delta)$ and of the Brzozowski derivatives of regular languages $L_i$.

## 3.2 The second-level analogy

Thus, at least on this equational level, DFAs (resp. their regular languages) are explicitly analogous to systems of linear differential equations (resp. their solutions); this analogy is not superficial than one might first think in that we can further develop this equational-level analogy to the higher level, which is a natural analogue of the categorical framework of the galois theory for linear differential equations [23, 24]; this higher-level analogy is also compatible with the first-level one. In fact, the classical Eilenberg theory [14] can now be seen as a DFA counterpart to the galois theory of linear differential equations.

Before proceeding to their categorical frameworks, it is essential to recall briefly what such galois theories (and Eilenberg theory) provided us: On the one hand, a characteristic result in the galois theory of linear differential equations (or *differential galois theory* for short) is a characterization of *Liouvillian extensions* $L$ of a differential field $K$, i.e. those differential-field extensions $L/K$ which can be obtained by extending the functions in $K$ by the operations of (i) integrals, (ii) exponentials $e^f$, and (iii) solutions to algebraic equations with coefficients in $K$; to be more specific, a (Picard-Vessiot) extension $L/K$ is Liouvillian in the above sense if and only if the identity component of its galois group $Gal(L/K)$ is soluble (cf. e.g. §1.5, [18] for more detail). This theory is, of course, a differential-equational analogue of the classical galois theory for fields, which also has an analogous result; to be specific, the classical result in this theory states that a (galois) extension $L/\mathbb{Q}$ can be obtained by extending the numbers in $\mathbb{Q}$ by the operations of (i) four arithmetic operations, and (ii) roots, if and only if its galois group $Gal(L/\mathbb{Q})$ is soluble. Moreover, as discussed in [23, 24], Eilenberg theory also has analogous results to these two theories. For instance, according to the theorem of Schützenberger [20][12], which is the origin of Eilenberg theory, a regular language $L$ is definable by the first-order fragment of Büchi's monadic second-order logic (MSO) over finite words if and only if the *syntactic monoid* $M(L)$ of $L$ contains only trivial subgroups (cf. e.g. [17] for more details). As these results indicate, the galois theories (and Eilenberg theory) equally provided us a method to reduce problems on some expressibility of solutions of differential equations, algebraic equations and those of DFAs (i.e. regular languages) to purely algebraic problems on groups (and monoids).

It is notable that there exist their respective duality principles behind these analogous results; in fact, the last two theories could now be unified by a single duality theorem for *semi-galois categories* in a precise sense discussed in [23, 24]; and this unification makes the above analogy between DFAs and linear differential equations a bit more formal. Technically, the differential galois theory can be

[12] Together with the works of McNaughton and Papart; see e.g. [17] for some historical background.

formulated in terms of the duality theorem between (pro-) algebraic groups (schemes) and (neutral) Tannakian categories [18]; the classical galois theory for field extensions (as well as its analogue for covering spaces) can also be formulated in terms of the duality theorem between profinite groups and galois categories [22]; and further Eilenberg theory can be formulated in terms of the duality theorem between profinite monoids and semi-galois categories [23, 24]. In the sense that our last duality theorem includes the second one (for galois theory), we may say that Eilenberg theory is the monoid variant of the classical galois theory; and in this relation, notable for our current subject is that Tannakian categories too are formally defined as the vector-space enriched variants of (semi-) galois categories.[13]

Therefore, at this stage, we observe that Eilenberg theory and differential galois theory share a common theoretical structure, despite of their ostensible difference; this comparison also highlights where these two theories are actually different. In particular it seems most characteristic that, while (semi-)galois categories become equivalent to those of representations of profinite groups (monoids) in *finite sets*, Tannakian categories become equivalent to those of representations of suitable compact groups in *finite-dimensional vector spaces*[14]; and this difference also appears in the fineness of the topology of their dual fundamental groups (or monoids) (cf. §4).

The geometric ingredient of this difference is best clarified when we consider concrete examples. Thanks to the fact that both classes of categories are defined axiomatically, it is often the case that a category $\mathscr{C}$, which consists of geometric objects and seemingly has nothing to do with (linear/finite set) representations of groups (monoids), is actually equivalent to a category of such representations. In the following we see how such categorical equivalences arise, by three geometric examples: The first example is a Tannakian category known in differential galois theory; the second one is also a related Tannakian category and intended for a comparison to the third one, which is the semi-galois category introduced by Borger and de Smit [5, 6, 7] and the main objective of our study in [25, 26].

**Example 3.2.1.** For instance, let $\mathscr{C}$ be the category of complex vector bundles with connections over a connected Riemann surface $S$, say, $S := \mathbf{P}^1 \setminus \{0, \infty\}$; then $\mathscr{C}$ forms a Tannakian category, and equivalent to the category $\mathrm{Rep}_{\pi_1(S)}$ of finite-dimensional complex representations of the fundamental group $\pi_1(S) = \pi_1(S, 1) \simeq \mathbb{Z}$. (See [18] for more detail.)

To be specific, this equivalence $\mathrm{F} : \mathscr{C} \to \mathrm{Rep}_{\pi_1(S)}$ is given by taking the *monodromy representations* of a system of differential equations arising from $(\mathcal{V}, \nabla)$: Indeed, by definition, each object of $\mathscr{C}$ is a pair $X = (\mathcal{V}, \nabla)$ of a vector bundle $\mathcal{V}$ over $S$ and a connection $\nabla : \mathcal{V} \to \Omega_S \otimes \mathcal{V}$, where $\Omega_S$ denotes the line bundle of holomorphic differential 1-forms on $S$. In our example $S = \mathbf{P}^1 \setminus \{0, \infty\}$, we have a global coordinate $z$ so that $\Omega_S = O_S dz$, where $O_S$ is the sheaf of holomorphic functions on $S$; and every $\mathcal{V}$ is free, i.e. $\mathcal{V} \simeq O_S^m$. Therefore, giving a connection $\nabla : \mathcal{V} \to \Omega_S \otimes \mathcal{V}$ is equivalent to giving the composition:

$$\nabla_{\frac{d}{dz}} : \mathcal{V} \to O_S dz \otimes \mathcal{V} \xrightarrow{\frac{d}{dz} \otimes 1} \mathcal{V}; \tag{3.16}$$

hence, if we take a basis $\{e_1, \cdots, e_m\} \subseteq \mathcal{V}(S)$ over $O_S$, we have $A = (a_{ij})$ with $a_{ij} \in O_S(S)$ so that $\nabla e_i = \sum_j a_{ij} dz \otimes e_j$, and thus, $\nabla(\sum_i f_i e_i) = \sum_{ij} (\frac{df_i}{dz} + a_{ij} f_j) dz \otimes e_i$ for $f_i \in O_S$; in other words,

[13] To be more precise, at least to the best of our knowledge, we do not yet know whether there exists a formally common framework that unifies (several variants of) Tannakian categories *and* (semi-) galois categories, except for the discussion of Bruguieres [8]; see also the work of Schäppi [19] for a unification of several Tannakian categories.

[14] There are several variants of Tannakian categories [19]. Also, in terms of the mythical *field with one element* $\mathbb{F}_1$, finite sets are considered as vector spaces (or affine spaces) over $\mathbb{F}_1$; therefore, at least in this mythical (or intuitive) sense, (semi-)galois categories can be informally considered as special cases of Tannakian categories, where the base field is the "smallest field" $\mathbb{F}_1$.

the operator $\nabla_{\frac{d}{dz}}$ is represented as $\nabla_{\frac{d}{dz}} = \frac{d}{dz} + A$ with respect to the basis $\{e_i\}$ on $\mathcal{V}(S)$; and $\nabla$ is determined by $A$.

Given $X = (\mathcal{V}, \nabla) \in \mathscr{C}$, we can define the sheaf $\mathcal{W}_X$ on $S$ of *solutions to the differential equation* $\nabla\xi = 0$ by $\mathcal{W}_X(U) := \{\xi \in \mathcal{V}(U) \mid \nabla\xi = 0\}$ for each open subset $U \subseteq S$. Then $\mathcal{W}_X$ forms a locally constant sheaf of complex vector spaces of the same rank $m$. As above, if $\{e_1, \cdots, e_m\}$ is a basis of $\mathcal{V}(S)$, the differential equation $\nabla\xi = 0$ is equivalent to the system of linear differential equations $\frac{df_i}{dz} + \sum_j a_{ij} f_j = 0$ $(i = 1, \cdots, m)$, or equivalently:

$$\frac{d}{dz}\begin{pmatrix} f_1 \\ \vdots \\ f_m \end{pmatrix} \quad = \quad -A\begin{pmatrix} f_1 \\ \vdots \\ f_m \end{pmatrix}. \tag{3.17}$$

By $a_{ij} \in O_S(S)$, we can choose $\{e_i\}$ so that $A$ is locally constant; and the local solution space $\mathcal{W}_p$ of this equation at each point $p \in S$ is of dimension $m$ over $\mathbb{C}$. Then each $\gamma \in \pi_1(S, 1)$ gives rises to a $\mathbb{C}$-linear automorphism $\rho_X(\gamma) : \mathcal{W}_1 \to \mathcal{W}_1$ by analytic continuation of local solutions along $\gamma$, hence a linear representation $\rho_X : \pi_1(S, 1) \to GL(\mathcal{W}_1)$. The equivalence $\mathrm{F} : \mathscr{C} \to \mathrm{Rep}_{\pi_1(S)}$ is given by this correspondence $X \mapsto \mathrm{F}(X) := (\mathcal{W}_1, \rho_X)$.

**Example 3.2.2.** Let $(K, \partial)$ be the differential field given by $K := \mathbb{C}(z)$ and $\partial := \frac{d}{dz}$; and let $\mathscr{C}$ be the category of pairs $(V, \nabla)$, where $V$ is a finite-dimensional vector space over $K = \mathbb{C}(z)$ and $\nabla$ is a *regular singular connection* $\nabla : V \to \Omega \otimes V$ *with singular locus in* $\{0, \infty\} \subseteq \mathbf{P}^1$ (cf. §6.4, [18]), where $\Omega := \mathbb{C}(z)dz$ denotes the meromorphic differential 1-forms on the Riemann sphere $\mathbf{P}^1 = \mathbb{C} \cup \{\infty\}$. Then $\mathscr{C}$ forms a Tannakian category, equivalent to the category $\mathrm{Rep}_{\mathbb{Z}}$ of finite-dimensional complex representations of the additive group $\mathbb{Z} = \pi_1(\mathbf{P}^1 \setminus \{0, \infty\}, 1)$. In particular, in relation to the next example of semi-galois categories, we are concerned with the result that every irreducible module $(V, \nabla) \in \mathscr{C}$ arises as the *generic fiber* of some vector bundle $(\mathcal{V}, \nabla)$ on $\mathbf{P}^1$ with a regular singular connection $\nabla$ (Theorem 6.22, §6.5, [18]).

**Example 3.2.3.** Let $K$ be a number field and $O_K$ the ring of integers in $K$; the semi-galois category $\mathscr{C}_K$ introduced by Borger and de Smit [6] consists of $\Lambda$-*rings finite etale over* $K$ *with integral models* (cf. §2, [25]). To highlight some analogy, we briefly recall basic concepts; for more details, the reader is referred to the original sources [6, 5, 2, 25, 26].

For a flat $O_K$-algebra $A$, a $\Lambda$-*structure* on $A$ is defined as a family $\{\psi_{\mathfrak{p}} : A \to A\}_{\mathfrak{p}}$ of $O_K$-algebra endomorphisms $\psi_{\mathfrak{p}} : A \to A$ indexed by the maximal ideals $\mathfrak{p} \in P_K$ of $O_K$ such that, for all $a \in A$ and $\mathfrak{p}, \mathfrak{q} \in P_K$:

1. $\psi_{\mathfrak{p}} a - a^{N\mathfrak{p}} \in \mathfrak{p}A$;
2. $\psi_{\mathfrak{p}}\psi_{\mathfrak{q}} = \psi_{\mathfrak{q}}\psi_{\mathfrak{p}}$;

where $N\mathfrak{p}$ denotes the absolute norm of $\mathfrak{p}$. The first condition says that $\psi_{\mathfrak{p}}$ is a Frobenius lift; and the second says that they commute with each other. A pair $(A, \psi_{\mathfrak{p}})$ of a flat $O_K$-algebra $A$ and a $\Lambda$-structure $\{\psi_{\mathfrak{p}}\}$ on $A$ is called a $\Lambda$-*ring*; in particular, if $X$ is a $K$-algebra, any commuting family $\{\psi_{\mathfrak{p}}\}$ of $K$-algebra endomorphisms on $X$ satisfies the two conditions, and the pair $(X, \psi_{\mathfrak{p}})$ forms a $\Lambda$-ring. In the sense we discussed in §2.1 [25], the $\Lambda$-structure can be seen as a kind of "differential structure" on $A$; see also [9, 10].

The category $\mathscr{C}_K$ is defined in [6, 5] as the one consisting of finite etale $\Lambda$-rings $X$ over $K$ having *integral models*; this means that $X$ has a sub $\Lambda$-ring $A \subseteq X$ finite over $O_K$ such that $X = K \otimes A$;

or in other words, $(\mathrm{Spec}(X), \psi_{\mathfrak{p}})$ is the generic fiber of $(\mathrm{Spec}(A), \psi_{\mathfrak{p}})$. It was shown by Borger and de Smit [5, 6] that the (opposite) category $\mathscr{C}_K$ forms a semi-galois category together with the fiber functor $\mathrm{F}_K : \mathscr{C}_K^{op} \to \mathbf{Sets}_f$ being given by taking the geometric points $\mathrm{F}_K(X) := \mathrm{Hom}_K(X, \bar{K})$ and the fundamental monoid $\pi_1(\mathscr{C}_K, \mathrm{F}_K)$ is isomorphic to the *Deligne-Ribet monoid* $DR_K$, the inverse limit of certain *ray class monoids* $DR_{\mathfrak{f}}$ ($\mathfrak{f} \in I_K$), cf. §2 [26]; most importantly, their proof clarified an intrinsic connection between classical class field theory and (generalized) Λ-rings [3], which have an origin in *special* $\lambda$*-rings* in Grothendieck $K$-theory (§1, [5]); moreover, we showed in [26] that the galois objects of $\mathscr{C}_K$ for imaginary quadratic fields $K$ admit natural analytic description in terms of deformation families of Fricke's modular functions $f_a$ ($a \in \mathbb{Q}^2/\mathbb{Z}^2$). In view of these results, it seems that the framework of (generalized) Λ-rings [3, 4] and their semi-galois categories $\mathscr{C}_K$ [5, 6, 7, 25, 26] provide us a right angle to look at classical class field theory.

In order to see that the Λ-structure on rings can be seen as a kind of differential structure, and to highlight some analogy with the above examples of Tannakian categories, we here rephrase the above construction of $\mathscr{C}_K$ in terms of the *arithmetic jet spaces* [4]; in fact, this can be phrased in terms of a more generic topos theory as we mentioned in §5.1 [25] too, which might be useful when one constructs other examples.

Let $\mathscr{E}$ be a (connected Grothendieck) topos and $\mathscr{D} : \mathscr{E} \to \mathscr{E}$ be a left exact comonad; a geometric example due to [4] is the topos $\mathscr{E}$ of *spaces over* $\mathrm{Spec}(\mathbb{Z})$ and the left exact comonad $\mathscr{D} : \mathscr{E} \to \mathscr{E}$ associating to each space $X \in \mathscr{E}$ the *arithmetic jet space* $\mathscr{D}(X)$ (cf. also [2]). Then (or equivalently), there is a surjective geometric morphism $\gamma_{\mathscr{D}} : \mathscr{E} \twoheadrightarrow \mathscr{E}_{\mathscr{D}}$ to the topos $\mathscr{E}_{\mathscr{D}}$ of $\mathscr{D}$-comodules $(X, \nabla)$, where we call the structure map $\nabla : X \to \mathscr{D}(X)$ as a *connection* on $X \in \mathscr{E}$. (We shall regard them as $\mathbb{F}_1$-analogues of connections $\nabla : \mathcal{V} \to \Omega \otimes \mathcal{V}$ on vector bundles $\mathcal{V}$; in the case of [2], $\mathscr{D}$-comodule strucrueres are equivalent to Λ-structures, cf. [2]; see also [10])

Given a $\mathscr{D}$-comodule $(B, \nabla) \in \mathscr{E}_{\mathscr{D}}$, we can define a category $\mathscr{C}_{(B,\nabla)}$ consisting of $\mathscr{D}$-comodule maps $(X, \nabla) \to (B, \nabla)$ such that $X \to B$ is locally finite and locally constant in $\mathscr{E}$, and $\mathscr{D}$-comodule maps over $(B, \nabla)$; denoting by $\mathscr{G}_B$ the galois category of locally finite and locally constant objects $X \to B$ over $B$ in $\mathscr{E}$, we have a natural forgetful functor $U : \mathscr{C}_{(B,\nabla)} \to \mathscr{G}_B$, which is by construction exact and reflects isomorphisms; $\mathscr{G}_B$ having a fiber functor $p^* : \mathscr{G}_B \to \mathbf{Sets}_f$ induced from a point $p : \mathbf{Sets} \to \mathscr{E}$ of $\mathscr{E}$, the composition $\mathrm{F} := p^* \circ U : \mathscr{C}_{(B,\nabla)} \to \mathbf{Sets}_f$ is exact and reflects isomorphisms so that the pair $(\mathscr{C}_{(B,\nabla)}, \mathrm{F})$ forms a semi-galois category (cf. Proposition 6, [25]).

The construction of $\mathscr{C}_K$ is rephrased in these topos-theoretic terms applying to the framework of Λ-algebraic geometry [2]. To be more specific, let $\mathscr{E}$ be the topos of spaces over $\mathrm{Spec}(O_K)$ (i.e. sheaves on affine schemes over $\mathrm{Spec}(O_K)$ with respect to the etale topology, cf. [4]); and $\mathscr{D} : \mathscr{E} \to \mathscr{E}$ be the comonad associating to each space $X \in \mathscr{E}$ its arithmetic jet space $\mathscr{D}(X) \in \mathscr{E}$. As the base $\mathscr{D}$-comodule $(B, \nabla) \in \mathscr{E}_{\mathscr{D}}$, we take $B = \mathrm{Spec}(K)$ together with the trivial Λ-structure $\nabla_K$ (i.e. $\psi_{\mathfrak{p}}$'s are all the identity), which induces a semi-galois category $\mathscr{C}_{(\mathrm{Spec}(K),\nabla_K)}$ as above. Nevertheless, this $\mathscr{C}_{(\mathrm{Spec}(K),\nabla_K)}$ itself is not equal to the target $\mathscr{C}_K$, i.e. $\mathscr{C}_K$ consists of those $(X, \nabla) \in \mathscr{C}_{(\mathrm{Spec}(K),\nabla_K)}$ which have *integral models*, in other words, arise as the generic fiber of some affine Λ-scheme finite over $\mathrm{Spec}(O_K)$.

**Remark 3.2.4.** It is notable that these duality theories for Tannakian categories and semi-galois categories equally play the role in reducing geometric objects (such as vector bundles $(\mathcal{V}, \nabla)$ with connections/finite etale Λ-rings over fields) to more concise objects that at least seemingly have less information than the original geometric raw data: In the case of the Tannakian category of vector bundles $(\mathcal{V}, \nabla)$ on Riemann surfaces with connections (Example 3.2.1), these geometric data could be classified (up to isomorphisms) by their monodromy representations $(\mathcal{W}_1, \rho_X)$, which seemingly contain quite local information on the fiber space $\mathcal{W}_1$; similarly, in the case of semi-galois category of finite etale $\Lambda$-rings over $K$ with integral models (Example 3.2.3) too, the $\Lambda$-rings $(X, \psi_{\mathfrak{p}})$ could be classified by their geometric fiber *sets* $\mathrm{F}_K(X)$ together with the action of $\psi_{\mathfrak{p}}$. The major difference is that, while the former needs linear-space data, the latter needs only finite sets (i.e. 0-dimensional data) for their classifications; we note that these distinction can be equally stated as the distinction of the class of geometry.

# 4 Comparisons

As mentioned in §1 and §2, physics and theory of formal languages and computation are still quite different disciplines in several respects; but as we observed in the above arguments (§3), it also seems true that their mathematical languages do share some common (or analogous) theoretical structure at least when restricted to their respective simplest cases, i.e. linear differential equations and finite automata. In view of the several aspects discussed in §2.3 concerning these two disciplines, it would be in order to discuss here some geometric and conceptual ingredients of these observations; it also seems reasonable to wonder whether these observations extend to higher computational models and non-linear differential equations (or in other words, more *contextual* systems). The latter problem (on the extendibility) will be discussed elsewhere; in this section, we focus on the former conceptual issue to summarize this paper.

On the whole, as briefly mentioned in §1 too, this paper tried to re-locate the classical theory of formal languages and computations coherently in a wider *geometric* context, aiming to rebuild this discipline so as to be comparable to physics in some mathematically formalized manner. Historically speaking, the studies on languages and computability concept are deeply relevant to understanding of the mechanism of our mind and logical thoughts; but in nature of the subject, we are immediately faced to several methodological challenges about which we were not bothered in physics: Indeed the objects of study in this discipline are our experiences in our internal reality, hence, inevitably subjective in nature. In this difficulty, formal language theory focused on the combinatorial structures of (formal) languages (i.e. sets of finite words, or texts) so that we can discuss linguistic issues in a more quantitative or qualitative manner with formally defined mathematical models, i.e. generative grammars (§2); this methodology was fruitful enough so that the paradigm has been influential to diverse disciplines, say, psychology, cognitive science, and computer science.

With this background, this paper was devoted to highlight some analogies between the simplest fragments of the mathematical languages traditional in theory of computation and physics, namely, DFAs (resp. regular languages) and linear differential equations (resp. their solutions). As discussed in §3, the analogies between them are multi-layered from an equational-level to a categorical level, and seem to be more than superficial. Conceptually, both (discrete/continuous) dynamical systems (i.e. DFAs/linear differential equations) have a common characteristic feature that their behaviors are both determined *locally*; these features best appeared in their equational descriptions (cf. §3.1); these analogies could be strengthened by the categorical axiomatizations of their traditional classification theories (i.e. Eilenberg theory and differential galois theory) in terms of the duality theories for semi-galois categories and Tannakian categories (cf. §3.2). In the informal sense that semi-galois categories can be regarded as the simplest case (or the Boolean-valued case) of Tannakian categories (i.e. those over the smallest field $\mathbb{F}_1$; cf. the footnote in the 12-th page), our categorical comparisons of the two classical theories re-locates them in a single context, clarifying where these two theories are actually different.

To be more specific, as mentioned in §3.2 too, it is characteristic that, technically speaking, fiber functors on (semi-) galois categories are *finite-set* valued, while those of Tannakian categories are *vector-space* valued; these difference was more clarified when we consider their geometric examples (Example 3.2.1, Example 3.2.3): The $\Lambda$-rings $X \in \mathscr{C}_K$ finite etale over a field $K$ could be classified (or determined) up to isomorphism only by the data of their fiber *sets* $\mathrm{F}_K(X)$, while vector bundles $(\mathcal{V}, \nabla) \in \mathscr{C}_S$ on Riemann surface with connections are classified (or determined) up to isomorphism by the data of their *linear space* $\mathcal{W}_1$ of local solutions. On the one hand the objects $X$ of the former semi-galois category $\mathscr{C}_K$ are geometrically 0-dimensional (generally, etale over the base), and thus, they could be determined only by their fiber sets; however, in the case of the objects $(\mathcal{V}, \nabla) \in \mathscr{C}_S$ in the latter Tannakian category, the local solution space $\mathcal{W}_1$ of the differential equation $\nabla\xi = 0$ is a linear space over the complex number field $\mathbb{C}$, and the analytic continuation along each $\gamma \in \pi_1(S, 1)$ defines a $\mathbb{C}$-linear automorphism on $\mathcal{W}_1$. In view of that it is already a non-trivial fact that the whole structure $X = (\mathcal{V}, \nabla)$ is determined only by such quite local information $(\mathcal{W}_1, \rho_X)$, it would be generally impossible in principle to reduce this linear-space data to further sparse data such as finite fiber sets for recovering the original geometric raw data $(\mathcal{V}, \nabla)$. In other words, according to the geometric structure/complexity of objects to be classified, necessary data to parametrize them (or classify them up to some identification) can vary; in some sense, the topology of necessary data of such parametrization represents their geometric complexity in a qualitative manner.

With these geometric comparisons in mind, let us come back to our consideration on linguistic and computational issues (§2). As mentioned above, formal language theory conventionally studies formal languages, i.e. mathematically, sets $L \subseteq \Sigma^*$ of finite words, which combinatorially model text data sets in natural language. Similarly, several computational models such as Turing machines too model our computational procedures (such as, *to write down some symbols on a paper, then read the symbols to process the computations*) with some sorts of discrete dynamical systems (i.e. dynamical systems whose state spaces have the discrete topology). In this way, the classical theory of formal languages and computations models our linguistic/logical activities with models of discrete kinds, while physics conventionally models the time evolutions of the universe with models of continuum kinds (i.e. their states are parametrized by real/complex numbers) such as symplectic manifolds or $C^*$-dynamical systems.

It seems that, nevertheless, the discrete nature of mathematical models traditional in the former discipline is not intrinsic in the subject itself; rather, it seems that such a topological sparseness of conventional models comes from the technological limitation (in the early days of this discipline) of measurements of (physical) quantities relevant to our linguistic and logical activities. In view of the recent development of neuro-science, this old technological limitation of measurements will be overcome sooner or later to develop new kind of linguistic/computational/logical models. (Actually there are already many of such neuroscience-based models.)

In any case, however, we will continue to use some mathematical language of *geometry*; and this is why we emphasized that we re-locate the classical theory of formal languages and computations in general geo-metric context (cf. §1); in fact, we regard the activity of developing a mathematical theory of physics (or any mathematical sciences) on the whole as the very linguistic activity, hence, the very scope of our *language theory* in an extended and unconventional sense, where all classes of geometry are our formal models of general languages, not limited to the discrete one. To highlight the point a bit more, remember that we said in §2.3 that we compare the traditional "*mathematical languages*" of physics and theory of computations rather than saying that we compare "physics and theory of computations" since the comparisons of *languages* (in our extended sense) are possible at mathematical level because it just concerns comparisons of geometries, hence, is purely the matter of geometry. Indeed, geometries (such as differential, non-commutative, discrete ones etc.) have been our formal languages to parametrize entities and their dynamics of concern in the external reality;

the only essential difference in these mathematical-scientific activities is in the kind of geometries (or mathematical languages) that we utilize. In this respect, comparisons of geometries constitute a notably large fragment of general language theory in our extended sense, which themselves would be mathematically formalizable within geometry[15].

# A Notations on DFAs and regular languages

Let $\Sigma$ be an arbitrary alphabet (finite or infinite). By definition, a language $L \subseteq \Sigma^*$ over $\Sigma$ is called a *regular language* if it is recognized by some *deterministic finite automaton* (DFAs for short) over $\Sigma$. To be precise, DFAs over $\Sigma$ are defined as follows:

**Definition A.0.1** (DFAs)**.** A *deterministic finite automaton* (DFA) over $\Sigma$ is a pair $\mathcal{A} = (S, \delta)$ of a finite set $S$ and a map $\delta : S \times \Sigma \to S$, where the elements of $S$ are called the *states* of $\mathcal{A}$ and the map $\delta : S \times \Sigma \to S$ is called the *transition function* of $\mathcal{A}$.

Given a DFA $\mathcal{A} = (S, \delta)$ over $\Sigma$, the transition function $\delta : S \times \Sigma \to S$ naturally extends to a map $\delta : S \times \Sigma^* \to S$ by induction on the length of finite words $w \in \Sigma^*$: namely, $\delta(s, \varepsilon) := s$ and $\delta(s, ua) := \delta(\delta(s, u), a)$ for $u \in \Sigma^*$ and $a \in \Sigma$; also, given an *initial state* $s_0 \in S$ and a set $F \subseteq S$ of *final states*, we can define the language $L(\mathcal{A}, s_0, F) \subseteq \Sigma^*$ as follows:

$$L(\mathcal{A}, s_0, F) \quad := \quad \{w \in \Sigma^* \mid \delta(s_0, w) \in F\}. \tag{A.1}$$

**Definition A.0.2** (regular language)**.** A language $L \subseteq \Sigma^*$ is called a *regular language* if there exist a DFA $\mathcal{A} = (S, \delta)$, initial state $s_0 \in S$ and a set $F \subseteq S$ of final states, we have $L = L(\mathcal{A}, s_0, F)$.

# References


[1] Samson Abramsky. What are the fundamental structures of concurrency? We still don't know! *Electric Notes on Theoretical Computer Science*, 162:37–41, 2006.

[2] James Borger. Lambda-rings and the field with one element, 2009. arXiv:0906.3146.

[3] James Borger. The basic geometry of Witt vectors I, the affine case. *Algebra and Number Theory*, 5(2):231–285, 2011.

[4] James Borger. The basic geometry of Witt vectors II, Spaces. *Mathematische Annalen*, 351:877–933, 2011.

[5] James Borger and Bart de Smit. Galois theory and integral models of $\lambda$-rings. *Bulletin of London Mathematical Society*, 40(3):439–446, 2008.

[6] James Borger and Bart de Smit. Lambda actions of rings of integers, 2011. arXiv:1105.4662.

[7] James Borger and Bart de Smit. Explicit class field theory and the algebraic geometry of $\lambda$-rings, 2018. arXiv:1809.02295v1.

[8] Alain Bruguieres. Dualite de galois-grothendieck et dualite tannakienne, 2013. `https://imag.umontpellier.fr/~bruguieres/docs/galois_tannaka_anniv_georges_2013.pdf`.


[15] Compare to the self-reference nature of the classical theory of computations.

[9] Alexandru Buium. Geometry of $p$-jets. *Duke Mathematical Journal*, 82(2):349–367, 1996.

[10] Alexandru Buium. *Foundations of Arithmetic Differential Geometry*. American Mathematical Society, 2017.

[11] Noam Chomsky. *Aspects of the Theory of syntax*. MIT Press, 1956.

[12] Noam Chomsky. *Syntactic structures*. Mouton and Co., B. V., Publishers, The Hague, 1957.

[13] Gilles Dowek. Around the Physical Church-Turing thesis: Cellular automata, formal languages, and the principles of quantum theory. In *LATA 2012*, pages 21–37, 2012.

[14] Samuel Eilenberg. *Automata, Languages, and Machines*. Academic Press, Inc., 1974.

[15] James Clerk Maxwell. *Theory of Heat*. Dover Books on Physics, 2001.

[16] Stephan Mertens. Computational complexity for physicists. *Computing in Sciences and Engineering*, pages 31–47, 2002.

[17] Jean-Eric Pin. Mathematical foundation of automata theory. Available from: `http://www.liafa.jussieu.fr/~jep/PDF/MPRI/MPRI.pdf`.

[18] Marius Van Der Put and Michael Singer. *Galois theory of linear differential equations*. Springer-Verlag Berlin Heidelberg, 2003.

[19] Daniel Schappi. The formal theory of Tannaka duality. *Asterisque*, (357):148, 2013.

[20] Marcel-Paul Scutzenberger. On finite monoids having only trivial subgroups. *Information and Control*, 8:190–194, 1965.

[21] Michael Sipcer. *Introduction to the Theory of Computation*. Course Technology Inc; International ed of 2nd revised edition, 2005.

[22] Tamas Szamuely. *Galois groups and fundamental groups*. Cambridge University Press, 2009.

[23] Takeo Uramoto. Semi-galois Categories I: The Classical Eilenberg Variety Theory. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science (LICS'16)*, pages 545–554, 2016.

[24] Takeo Uramoto. Semi-galois Categories I: The Classical Eilenberg Variety Theory, 2017. arXiv:1512.04389v4 (Extended version of LICS'16 paper).

[25] Takeo Uramoto. Semi-galois Categories II: An arithmetic analogue of Christol's theorem. *J. Algebra*, 508(2018):539–568, 2018.

[26] Takeo Uramoto. Semi-galois Categories III: Witt vectors by deformations of modular functions, 2020. arXiv:2007.13367v7.

[27] Eugene Wigner. The unreasonable effectiveness of mathematics in the natural sciences. *Communications in Pure and Applied Mathematics*, 13(1):1–14, 1960.

[28] Yoshitaka Yamamoto. *Historical Evolution of thermodynamics philosophy (Japanese)*. Chikuma Shobo, 2008.