Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 18 (2013) 90 – 99

Procedia Computer Science

International Conference on Computational Science, ICCS 2013

Non locality, topology, formal languages: new global tools to handle large data sets

Emanuela Merelli[a,*], Mario Rasetti[b]

[a]School of Science and Technology, University of Camerino, Via del Bastione,1, Camerino 62032, italy
[b]ISI Foundation, Via Alassio 11-C, Torino 10126, Italy

Abstract

The basic idea that stems out of this work is that large sets of data can be handled through an organized set of mathematical and computational tools rooted in a global geometric vision of data space allowing to explore the structure and hidden information patterns thereof. Based on this perspective, the objective is naturally that of discovering and letting emerge, directly from probing the data space, the manifold hidden relations (patterns), e.g. correlations among facts, interactions among entities, relations among concepts and formally describing, in a semantic mining context, the discovered information. In this note, we propose an approach that exploits topological methods for classifying global information into equivalence classes and regular languages for describing the corresponding automaton as element an of hidden complex system.

Keywords: Topology of data; Mapping Class Group; Formal Language; Complex systems.

1. Introduction

Probably the most important fact in modern science is the dramatic change in paradigms that has seen reductionism challenged by holism. Complex systems can be defined as systems composed of many non-identical elements, entangled in loops of non-linear interactions. The challenge is to control the collective emergent properties of these systems, from knowledge of components to global behavior. A typical feature of complex systems is in fact emergence of non-trivial superstructures that cannot be reconstructed by applying a reductionist approach. Not only do higher emergent features of complex systems arise out of the lower level interactions, but the patterns that they create react back on those lower levels. We can consider a complex system made by two levels of information, the local information - i.e. the network of

*Corresponding author: tel.: +39-338-399-0412 ; fax: +39-0737-40-2561
E-mail address: emanuela.merelli@unicam.it.

Selection and peer review under responsibility of the organizers of the 2013 International Conference on Computational Science
doi:10.1016/j.procs.2013.05.172

interactive elements - and the global information - the emergence of global properties, possibly unknown, from the observed phenomenon. To construct a theory that allows to define and manage complex systems we need to reach forward to a real theory of complex systems, bearing on complex phenomena and data. We believe that one way to create such models is that of extracting them from the data by which the complex system itself is characterized. Handling large sets of data, understanding what kind of phenomena are hidden and trying to model the dynamics of the corresponding complex system, is a very ambitious goal whose output will contribute for reliable predictions. To this aim, in recent years, an integrated set of methods and concepts has emerged among which those, topology based, that will be introduce in this note. The seminal work of a number of authors, such as Carlsson [1], Edelsbrunner and Harer [2] and others introduced the basic idea that large sets of data can be handled only through a global geometric vision of data: the notion that it should be possible to incorporate data in a global topological setting, the 'space of data' – defined as a suitable collection of finite samples taken from the data set – and explore then the structure and hidden information patterns thereof [5]. Based on this perspective, the ultimate objective is naturally that of discovering and letting emerge, directly from probing the data space, the manifold hidden relations (patterns) that exist as correlations among events/facts, interactions among actors/agents or even relations among concepts, and semantically interpreting them as global properties associated to the mining context.

In this note, after reviewing the basic ingredients of topology at the basis of the proposed approach and pointing out its strengths and weaknesses, we intend to show that the proposed topological approach leads to the possibility of classifying data – even apparently disordered and noisy data sets – into equivalence classes with respect to certain global transformations of the data space, by the equivalent of what in topology is referred to as the *mapping class group*. There emerges an unexpected structure of the data set, which has quite a far-reaching interpretation in terms of formal language theory, and endows the semantics generated by the mining process with a new, powerful, efficient tool. The latter promises to play a role in the evolution/elaboration process leading from data to information, from information to knowledge and eventually from knowledge to wisdom.

2. The Proposed Approach: three main steps

The approach proposed in this note consists of constructing a global object, analysing the behaviour of the object, and describing the global object in a semantic domain.

Construction a global object

Three basic ideas provide the pillars over which the global, topological approach to data space is based: i) It is convenient to interpret the huge set of 'points' that constitute the space of data resorting to a family of simplicial complexes, parametrized by some suitably chosen 'proximity parameter'. It is this operation that converts the data set into a global topological object. In order to fully exploit the advantages of topology, the choice of such parameter should be metric independent, in general the expression of a "relation". ii) One can fruitfully deal with such topological complexes by the tools of algebraic and combinatorial topology. Specifically, the most efficient tool is the theory of persistent homology, appropriately adapted to the parameterized families of simplicial complexes characterizing the space of data when explored at

various proximity levels. This allows us to get rid in some way of the noise affecting the data considered. In our context, the reduction of noise is the result of the parametrized persistent homology. iii) It is possible and efficient to encode the data set persistent homology in the form of a parameterized version of topological invariants, in particular Betti numbers, i.e., the invariants representing the dimensions of the homology groups. These three steps provide an exhaustive knowledge (possibly approximate, if the cluster of points considered does not coincide with the entire space) of the global features of the space of data, even though such space is neither a metric space nor a vector space [5].
Given a space of data $\mathfrak{S}$, unordered collection of data represented just as a set of points, this first step consists in selecting, by the appropriate notion of proximity, a subgroup of data $\mathfrak{K} \subseteq \mathfrak{S}$. The topological information contained in $\mathfrak{K}$ is global and is coded in a set of topological invariants that summarizes the information over domains of parameter values, of the topological objects constructed in a discrete space from data. What we call *topological object* is a piece of information extracted from a set of data to which we aim to associate a meaning as much as possible coherent, coded in the global features of the space of data considered as topological space. As an example, suppose to have a space of data regarding some reality whatsoever (unknown), and suppose having identified a subset of data whose global information tells you that the topological object, call it CH_4, can be parameterized by a set of parameter values that range by $[w = 0.2715, w = 0.2453, w = 0.2389, w = 0.1513]$ and that the object is of genus $g = 3$ when $w = 0.2453$ and genus $g = 9$ when $w = 0.2389$.

Analysing the behaviour (properties) of the object
The second step consists of analysing the behavior of the topological object under all possible, topology preserving, transformations. This leads to classifying the space of data into classes of equivalence, each of which represents symmetries and regularities hidden within the space of data itself. They are determined by the cosets of the *mapping class group* of the topological space which is our object of analysis. The genus tells us by what kind of manifold such object is represented. While the group is the mathematical tool for constructing languages. Any group is presented by the set S of its generators and the set R of its relations. An example is given by the group G_{168}, which provides a basis for a surface of genus $g = 3$. This means that G_{168} is the basic ingredient by which to generate the languages suitable to describe any objects represented by a genus three 2-manifold. Even if elements in different classes may represent the same concept, the configuration with which they are spatially connected must be different; that means the order in the relations is changed. Going back to the above example of CH_4, if we take the topological object of genus $g = 3$ and we use G_{168} we can create by it the classes of equivalence of all transformations of CH_4, each class representing different spatial configurations.

Describing the global object
Once we have extracted the global object and generated its classes of behavioral equivalences, the third step consists of interpreting the object through the generation of formal languages and mappings into semantic domains. We recall that within the theory of formal languages there are two very meaningful results, the first asserts that for any *regular language* we can define a non-deterministic finite state automaton and for any non-deterministic finite state automaton

we can define a *determinist finite state automaton*. The second, by Myhill-Nerode[1], states that a language is regular if the strings of the language can be classified in a finite number of classes of equivalences and the number of equivalence classes is equal to the number of states of the minimal deterministic finite automaton accepting the language. Since both groups, the basis and the mapping class group determine, both in the manifold and over the space of data, a finite set of classes of equivalence, we can define two deterministic automata, one that recognizes the language of the reference basis and the other that recognizes the language of the space of data. In such a way we define two languages; a common language, in the above analogy the relational algebra, and a specific language. The idea is to use relational algebra to describe the relations hidden in the space of data. Referring to the CH_4 example, we can say that the common regular language is the language associated to G_{168} and the specific language is the language that described the relations, e.g. the hidden global properties of CH_4. What happens if we try to associate to these two language a semantic domain? First we discover that the space of data contains data *possibly similar* to those stored from the simulation of *methane molecules* (Fig. 1 of the Pascucci 's et. al. work illustrates methane electron iso-density surfaces [8]), then we discover that all the CH_4 molecules belongs to the structural isomer, because they are arranged in a unique spatial configuration where each hydrogen bond bonds with a single location on the carbon atom and there is no way to rearrange the hydrogen atoms.

2.1. Introduction to Topology of Space of Data

In order to better comprehending the scheme, it is necessary to recall that the homology is a mathematical tool that “measure” the shape of an object. The result of this measure is an algebraic object, a succession of groups. Informally, these groups encode the number and the type of “holes" in the manifold. A basic set of invariants of a topological space X is just its collection of homology groups, $H_i(X)$. Computing such groups is certainly non-trivial, even though efficient, algorithmic techniques are known to do it systematically. Important ingredients of such techniques, but also output of the computation, are just Betti numbers; the i-th Betti number, $b_i = b_i(X)$, denoting the rank of $H_i(X)$. It is worth remarking that Betti numbers often have intuitive meaning: for example, b_0 is simply the number of connected components of the space considered, while oriented 2-dimensional manifolds are completely classified by $b_1 = 2g$, where g is the genus (i.e., number of “holes”) of the manifold, so as b_2 classifies the 3-dimensional and b_n the n-dimensional manifolds. What makes them convenient is the fact that in several cases knowing the Betti numbers is the same as knowing the full space homology. Sometimes to know the homology groups it is sufficient to know the corresponding Betti numbers, typically much simpler to compute. In the absence of torsion, if one wants to distinguish two topological objects via their homology, their Betti numbers may already do it. We already mentioned that data can be represented as unordered sequence of points in a n-dimensional space E_n, the ‘space of data’. Such space is typically not a vector space [indeed, every point of it is represented as a vector, i.e., a string of numbers in some field, but the ‘components’ of such vector have no meaning], and – even more manifestly – there is no reason to consider it Euclidean, as it is instead often done. All crucial information about the system

[1] Nerode, Anil (1958), “Linear Automaton Transformations”, Proceedings of the AMS 9

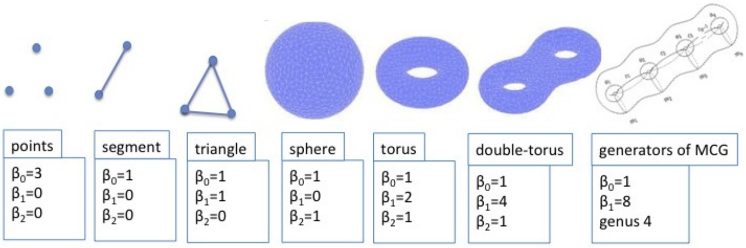

Fig. 1. Betti numbers and generators of *MCG*

the data in E_n refer to cannot be encoded in the global 'structure' of the data space, through its inherent, typically hidden, correlation patterns. The latter is what contains (and may provide) the relevant knowledge about the underlying phenomena which data represents.

The obvious conventional way to convert a collection of points within a space such as E_n into a global object is to use the point cloud as vertex set of a combinatorial graph, $\mathfrak{G}$, whose edges are exclusively determined by a given notion of 'proximity', specified by some weight parameter δ. This is a delicate point of the theory, because δ should not fix a 'distance', that would imply fixing some sort of metric, but rather provide information about 'dependence', i.e., correlation or, even better, relation. In case such dependence had to do with the distance, it should be a non-metric notion (for example, chemical distance, ontological distance). A graph of this sort, while capturing pretty well connectivity data, essentially ignores a wealth of higher order features beyond clustering. Such features can instead be accurately discerned by thinking of the graph as the 'scaffold' of a different, higher-dimensional, richer (more complex) discrete object, generated by completing the graph $\mathfrak{G}$ to a simplicial complex, $\mathfrak{K}$. The latter is a piecewise-linear space built from simple linear constituents (simplices) identified combinatorially along their faces. The decisions as how this is done, implies a choice of how to fill in the higher dimensional simplices of the proximity graph. Such choice is not unique, and different options lead to different global representations. Two among the most natural and common ones, equally effective to our purpose, but with different characteristic features, are: i) the Čech simplicial complex, where k-simplices are all unordered $(k+1)$-tuples of points of the space E_n, whose closed $\frac{1}{2}\delta$-ball neighborhoods have a non-empty mutual intersection; ii) the Rips complex, an abstract simplicial complex whose k-simplices are the collection of unordered $(k+1)$-tuples of points pairwise within distance δ. The Rips complex is maximal among all simplicial complexes with the given 1-skeleton (the graph), and the combinatorics of the 1-skeleton completely determines the complex. The Rips complex can thus be stored as a graph and reconstructed out of it. For a Čech complex, on the contrary, one needs to store the entire boundary operator, and the construction is more complex; however, this complex contains a larger amount of information about the data space toplogical structure.

Algebraic topology provides a mature set of tools for counting and collating holes and other topological pattern features, both spaces and maps between spaces, for simplicial complexes. It is therefore able to reveal patterns and structures not easily identifiable otherwise. As persistent homology is generated recursively, corresponding to an increasing sequence of values of δ.

Complexes grow with δ. This leads us to naturally identifying the chain maps with a sequence of successive inclusions. Persistent homology is nothing but the image of the homomorphism thus induced. The available algorithms for computatiing persistent homology groups focus typically on this notion of filtered simplicial complex. Most invariants in algebraic topology are quite difficult to compute efficiently. Fortunately, homology is exceptional under this respect because the invariants arise as quotients of finite-dimensional spaces.

3. Transformations of the Space of Data

Turning the space of data into a topological global object, as we do representing/approximating it by a (parametrized family of) simplicial complexes, allows us to consider its behavior under global topological transformations. Such transformations classify subspaces of (orbits in) E_n into equivalence classes.

The *mapping class group*, $\mathcal{G}_{\mathcal{MC}}$, is one of such sets (indeed a group) of transformations [3]. In order to discuss it, one has to consider two fundamental objects attached to (all) the 2-dimensional submanifolds $\mathfrak{K}$ of $\mathfrak{S}$: a group and a space. How these two objects relate to each other is crucial to understand what happens.

In terms of our problem, the space $\mathfrak{S}$ can represent the space of data, the topological global object, while the group $\mathfrak{K}$ is the tool that allows us to select the classes of equivalence by classifying the transformations and discoverying relations possibly hidden in the space of data. In order to make definitions and properties more clearly understandable, let's first define things as if $\mathfrak{S}$ were continuous. The group $\mathcal{G}_{\mathcal{MC}}$ is then defined to be the group of isotopy classes of orientation preserving diffeomorphisms of $\mathfrak{S}$ (that restrict to the identity on the boundary $\partial\mathfrak{S}$, if $\partial\mathfrak{S}$ is not empty): $\mathcal{G}_{\mathcal{MC}}(\mathfrak{S}) \equiv \mathrm{Diff}(\mathfrak{S})/\mathrm{Diff}_0(\mathfrak{S})$, where $\mathrm{Diff}(\mathfrak{S})$ is the group of diffeomorphisms of $\mathfrak{S}$, whereas $\mathrm{Diff}_0(\mathfrak{S})$ is the group of diffeomorphisms of $\mathfrak{S}$ isotopic to the identity, i.e., homotopic to the identity by a homotopy that takes the boundary into itself. $\mathcal{G}_{\mathcal{MC}}(\mathfrak{S})$ is generated by *Dehn's twists*. In the case of a (closed, orientable) Riemann surface $\mathfrak{S}$ of genus $g \geq 2$, for Γ a simple closed curve in $\mathfrak{S}$, and $\mathcal{A}$ be an annulus, i.e., a tubular neighborhood of Γ, the Dehn twist τ is the map from $\mathfrak{S}$ to itself which is the identity outside of $\mathcal{A}$ and inside corresponds to a full (2π) rotation of the boundaries of $\mathcal{A}$ – topologically equivalent to circles – with respect one to the other. A set of theorems of Dehn, Lickorish and Humpries prove that the minimal number of curves necessary to generate $\mathcal{G}_{\mathcal{MC}}(\mathfrak{S})$ is $2g + 1$ for $g > 1$. Typically curves Γ_j, $j = 1, \ldots, 2g + 1$, are chosen to be elements of the homology basis, i.e., representative cycles of the homology, of $\mathfrak{S}$.

In general the problem of finding the presentation of $\mathcal{G}_{\mathcal{MC}}(\mathfrak{S})$ requires the introduction of the appropriate combinatorial structure, which resides in the Hatcher-Thurston complex [4].

In group theory, one method of defining a group G is by its presentation $G \sim \langle S \mid R \rangle$. In short, the mapping class group of a topological (simplicial complex) space $\mathfrak{S}$ is the group of isotopy-classes of automorphisms of $\mathfrak{S}$; that is the group of all transformations of the space into a topologically identical object. Performing these transformations implies imposing an order on space $\mathfrak{S}$ and determine a set of equivalence classes that represent a partition of $\mathfrak{S}$. The mapping class group is presented by a set of generators and relations among generators that characterized the equivalence classes. Such presentation can be straigthforwardly expressed in the langugae generated by the group that provides the basis to which $\mathfrak{S}$ can be referred.

3.1. The $\mathcal{G}_{MCG}$ based on modular group $\mathfrak{Mod}$

As an application, we illustrate here an example in which the 'surface' $\mathfrak{S}$ is represented in a basis derived from the modular group – a feature that is 'universal' and can therefore be assumed with no loss of generality as generic. The *modular group* $\mathfrak{Mod}$ is isomorphic to a discrete group, the projective special linear group $PSL(2,\mathbb{Z})$. $\mathfrak{Mod}$ is the group of 2×2 matrices with integer entries and unit determinant, acting as a group of transformations.

$$\mathfrak{Mod} = \left\{ M = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \middle| a, b, c, d \in \mathbb{Z}\, ;\, ad - bc = 1 \right\}.$$

$\mathfrak{Mod}$ has presentation $\mathfrak{Mod} \sim \langle U, V |\, V^2, (UV)^3 \rangle$ where the generators are

$$V = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix},\ U = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}.$$

The modular group $\mathfrak{Mod}$ has a *principal congruence of invariant subgroups* $\mathfrak{Mod}_p \sim PSL(2,\mathbb{Z}_p)$, $p =$ odd prime, defined by $\mathfrak{Mod}_p = \left\{ M_0 \in \mathfrak{Mod} \middle| M_0 = \begin{pmatrix} \pm 1 & 0 \\ 0 & \pm 1 \end{pmatrix} (\mathrm{mod}\, p) \right\}$.

The factor $G_\omega \doteq \mathfrak{Mod}/\mathfrak{Mod}_p$, is a finite group of order ω, a Sylvester graph (*Lattice*) Σ_p, embedded in a manifold $\mathfrak{S}$ of genus g where $\omega = \frac{1}{2}p\,(p^2-1)$ represents both the number of elements of G_ω and the number of points of lattice Σ_p, and $g = \frac{1}{4!}\,(p+2)(p-3)(p-5)$. In other words, the Sylvester graph Σ_p is nothing but the orbit under G_ω of an arbitrary point in the canonical fundamental region of $\mathfrak{S}$.

As an example, take G_{168}, a basis for surfaces of genus g=3, and its presentation $G_{168} \sim \langle U, V \,|\, V^2, (UV)^3, U^7, (VU^4)^4 \rangle$ (notice that the action of each generator is assumed to be invertible, therefore – even though never explicitly done – together with U and V, the inverses U^{-1} and V^{-1} should be in principle listed in the presentation). Euler's theorem shows that graph Σ_7 has 24 heptagonal and 56 hexagonal plaquettes; each heptagon being surrounded by 7 hexagons. As the group manifold has genus > 0, the group is finite, a *global* relation appears, $(VU^4)^4 = \mathbb{I}$, which guarantees the closure of the *homology* of $\mathfrak{S}$ in this g=3 case. Locally the presence of heptagonal plaquettes implies that the surface exhibits negative curvature, the manifold is hyperbolic. Lattice Σ_7 is obtained from the hyperbolic disk shown in Fig.2 (B) by selecting an arbitrary point P on any of its triangular domains and finding its orbit under the whole G_{168} shown in Fig.2 (A), and then folding the resulting structure. Notice that the global topology of the set of points of Σ_7 provides the basis for any set of points with Betti number b_1=6. Fig.2 (C-F) shows how to obtain $\mathfrak{S}$ by folding of the 14-sided polygon.

It may be useful to note that the canonical homology basis of $\mathfrak{S}$ can be written in terms of words in the group relators U and V of G_{168}, that in turn can be used to express the mapping class group generators in a manner evidencing the underlying (non abelian) lattice structure.

A choice for such a representation of the canonical homology basis is given by

$$\begin{aligned} \mathbf{a}_1 &= (VU^3)^4 \quad , \quad \mathbf{b}_1 = U^{-1}(VU^3)^4U\ , \\ \mathbf{a}_2 &= U^{-1}(VU^4)^3VU(VU^4)^3VU \quad , \quad \mathbf{b}_2 = (VU^3)^3VU(VU^3)^3U^3(VU^3)^2VU\ , \\ \mathbf{a}_3 &= (VU^3)^3U^3(VU^3)^3U^3 \quad , \quad \mathbf{b}_3 = U^{-1}(VU^3)^3VU(VU^3)^3U^3(VU^3)^2VU^2\ , \end{aligned}$$

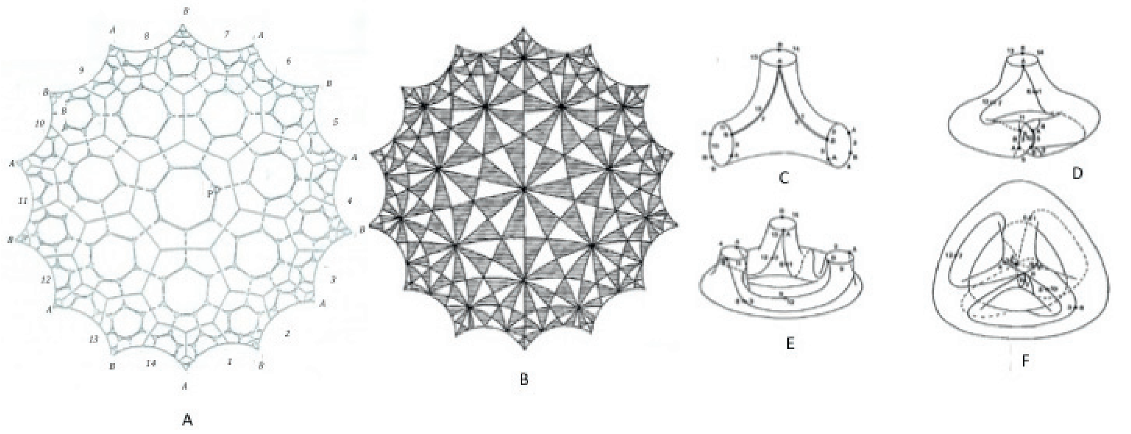

Fig. 2. A. Poincarè disk D_3 of the hyperbolic space $\mathfrak{G}$. B. Lattice Σ_7 as triangulation of $\mathfrak{G}$. F. Surface of genus g=3

which has been determined to respect the intersection form. This is done starting with the choice of a point P on the Poincaré disk D_3 (Fig. 2 (A)) and by drawing first a path $\mathbf{a}_1$, moving through the lattice with the U and V generators. The given path $\mathbf{a}_1$ starts in P, crosses the 5-th edge of the 14-th sided polygon and, due to the identification rule $(2\nu+1) \to (2\nu+6)[mod\ 14]$, reenters it through the 10-th side and closes in P. The cycle $\mathbf{b}_1$ then must have only one intersection with $\mathbf{a}_1$ and can be taken to start in P, exit through side 7 and reenter through side 12 to close in P. The cycle $\mathbf{a}_2$ is then drawn in such a way as to have no intersection with either $\mathbf{a}_1$ or $\mathbf{b}_1$, and from P exits through side 6, reappears in 1 to exit again trough 12, and reenters in 7 to join P. Furthermore $\mathbf{b}_2$, which must intersect $\mathbf{a}_2$ only, is made of three branches: one between sides 6 and 5 (which passes through P), the others between 10 and 9, and 14 and 1. Finally, $\mathbf{a}_3$ has a branch 2–5 and one 10–11 while $\mathbf{b}_3$ has branches 8–7, 12–11 and 2–8. Clearly, (shorter) alternative words corresponding to equivalent paths can be written by making use of group local relators $U^7 = V^2 = (UV)^3 = \mathbf{1}$ of G_{168}. Generators of mapping class group $\mathcal{G}_{MC}(\mathfrak{S}_3)$ of Σ_7 (or, better, of $\mathfrak{S}_3$) are the set of Dehn's twists around cycles $\mathbf{a}_i, \mathbf{b}_i, i = 1, 2, 3$.

4. Interpretations of the Space of Data

Considering all the transformations that a topological space undergoes and classifying them in classes of equivalences – via the mapping class group – allows us to define a regular language L that describes the data belonging to the partitions of space of maps between data – by applying known properties of formal language. We recall that given a finite set S, the free group is indicated by $\langle S \rangle$ consists of all the words that can be constructed with S; let $R \subseteq \langle S \rangle$ be the subset of the group consisting of words of S, $\langle S|R \rangle$ the biggest quotient group of $\langle S \rangle$ such that each element of R is identified by the identity, S the finite set of generators of the group $\langle S \rangle$, R the finite set of relators that is equivalence relations among elements of S and the presentation of a group G, $\langle S|R \rangle$, it is the free group $\langle S \rangle$ subject to the set of relations R.

Moreover recall that, if Σ is an alphabet, i.e., a finite non-empty set of symbols, a string over Σ is a finite sequence of symbols obtained by juxtaposition, the length of a string is the number of its symbols, the concatenation of two strings is the juxtaposition of the two strings, Σ^* is the set of all possible strings obtained over Σ and $L \subseteq \Sigma^*$, then L is a language. Σ^* can be defined by $\Sigma^* = \Sigma^0 \bigcup \Sigma^1 \bigcup \Sigma^2 \bigcup ...$ where Σ^i is the set of all strings whose length is i and $\Sigma^0 = \{\epsilon\}$, the language containing only the empty word (or string) ϵ. Given an alphabet $\Sigma = \{a, b, c \ldots\} \cup \{\epsilon\}$ the set of regular expressions is defined by $E ::= a \,|\, E + E \,|\, E \bullet E \,|\, E^* \,|(E)$ with $a \in \Sigma$, the set of regular languages are defined by $L[\epsilon] = \{\epsilon\}$; $L[a] = \{a\}$; $L[E + F] = L[E] \cup L[F]$; $L[E \bullet F] = L[E] \bullet L[F]$; $L[E^*] = (L[E])^*$. As an example, given an alphabet $\Sigma = \{a, b\} \cup \{\epsilon\}$ and a regular expression $E = ab^*$ the language $L[E] = L[a] \bullet L[b^*] = \{a\} \bullet \{b\}^* = \{a\}(\{\epsilon\} \cup \{b\} \cup \{bb\} \cup ...) = \{a, ab, abb, abbb, ...\}$ is the regular language that describes all the strings that start with 'a' and end with a certain number of 'b', possibly zero.

Thus, if we consider $\langle S \rangle$ be equivalent to Σ^*, $R \subseteq \Sigma^*$ is equivalent to a language L over Σ. Furthermore, if the quotient group $\langle S|R \rangle$ is obtained grouping similar elements by equivalence relations. Each generator belonging to the set R, gives rise to a class of equivalence whose elements satisfy an equality relation. Since the number of relations is finite, the corresponding classes of equivalence are also finite, then we can apply the two following well-known theorems that ensure the existence of the automaton accepting the language L and guarantee that is the smallest monoid that recognizes the language L.

Lemma 1. *Let S be a nonempty set and let $\sim$ be an equivalence relation on S. Then, $\sim$, yields a natural partition of S, where $\bar{a} = \{x \in S \,|\, x \sim a\}$. $\bar{a}$ represents the subset to which a belongs to. Each cell $\bar{a}$ is an equivalence class.*

Theorem 1 (Myhill-Nerode). *If L is any subset of Σ^*, one defines an equivalence relation $\sim$ (called the syntactic relation) on Σ^* as follows: $u \sim v$ is defined to mean $uw \in L$ if and only if $vw \in L$ for all $w \in \Sigma^*$. The language L is regular if and only if the number of equivalence classes of $\sim$ is finite. If a language is regular, then the number of equivalence classes is equal to the number of states of the minimal deterministic finite automaton A accepting L.*

Given the finite presentation of the group $G \sim \langle S|R \rangle$, with $S = s_1...s_m$ and $R = \{r_1, r_2...r_n\}$ we can associate to each relation $r_i \in R$, for $i = 1...n$, a language L_{r_i} that recognizes all the elements subject to r_i. The language L associated to the presentation G is the union of all languages that recognize all the relations in R whose symbols are in $\Sigma = S$. $L = \bigcup_{r_i, r_i \in R} L_{r_i}$

Finally, we recall that the expressive power of all the following formalisms is equal:
regular grammars (RG) → regular expressions (RE) → non deterministic finite state automata (NFA) → deterministic finite state automata (DFA). This property allow us to define a NFA and assert the existence of a FSA equivalent.

Following the example of G_{168} and its presentation $G_{168} \sim \langle U, V \,|\, V^2, (UV)^3, U^7, (VU^4)^4 \rangle$, the non deterministic finite state automaton shown in Fig 3 recognizes the language L of all the strings generated by the generators of group G_{168}. For the sake of readability, we label an edge of an automaton with a word as a shorthand for a sequence of states and transitions such that the concatenation of the labels on the transitions equals the word.
The language L_{168} defined over the alphabet $\Sigma = \{T, V\}$ for the presentation of the group $G_{168} = \langle U, V \,|\, V^2, (UV)^3, U^7, (VU^4)^4 \rangle$ is $L_{168} = L_{V^2} \bigcup L_{(UV)^3} \bigcup L_{U^7} \bigcup L_{(VU^4)^4}$.

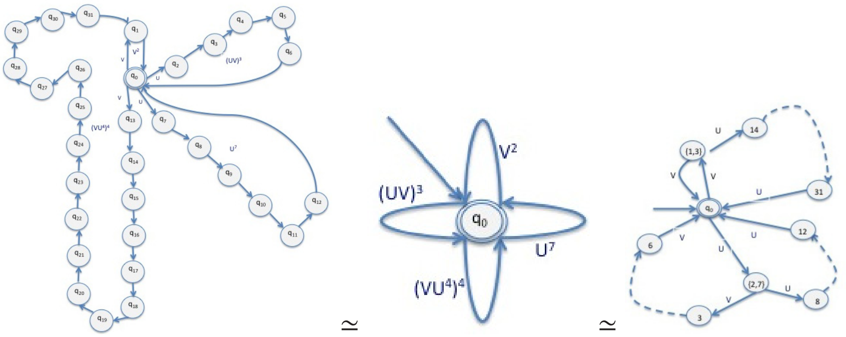

Fig. 3. NFA, shortened representation and equivalent DFA accepting $L \sim G_{168} = \langle U, V | V^2, (UV)^3, U^7, (VU^4)^4 \rangle$.

The language L_{168} allows to describe the *global* relations underlying the space of data whose surface is of genus g=3. Any point of the space is equivalent to a path, to a word of the language whose class of equivalence is an emergent pattern.

5. Conclusions

In this notes, we have introduced a new approach for analyzing a space of data that leads to the definition of a formal language supporting the interpretation of the space of data. The approach, topology-based, is able to process the data in a uniform way - through the filtration by persistent homology - but also characterize the space of data by different invariants so to emphasize different features (e.g., scales). We recall that topology has been widely use for multiscale analysis in the context of quantum gravity and theory of turbulence. Moreover, the use of topology for modeling multilevel complex systems is still a challenge that a specialized community of researchers is tacking with different approaches, among which those proposed in TOPDRIM project (www.topdrim.eu) [7].

Acknowledgements

We acknowledge the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme (FP7) for Research of the European Commission, under the FP7 FET-Proactive Call 8 - DyMCS, Grant Agreement TOPDRIM, number FP7-ICT-318121.

References

[1] G. Carlsson, *Topology and data*, Bulletin of the American Mathematical Society 46 (2) (2009) 255–308.

[2] H. Edelsbrunner, J.L. Harer, *Computational Topology*, AMS, 2009

[3] B. Farb, D. Margalit *A primer on Mapping Class Group*, Princeton University Press, 2011

[4] A., Hatcher, W., Thurston, *Topology*,19(221) 1980

[5] P. Lum, et al. *Extracting insights from the shape of complex data using topology*, Nature 3(1236), 2013

[6] J.E. Hopcroft, J. Ullman *Introduction to Automata Theory, Languages, and Computation*, Pearson Edu, 2000

[7] E. Merelli, M. Rasetti. *The Immune System as a metaphor for topology driven patterns formation in complex systems* In Proc. of 11th Int. Conf. ICARIS 2012. Springer LNCS 7597.

[8] V. Pascucci, K. Cole-McLaughlin *Efficient Computation of the Topology of Level Sets*. IEEE Visual. 2002.