A graph perspective on the genus of regular languages

Florian Deloup[2] **Guillaume Bonfante**[1]
1- Université de Lorraine - LORIA - Nancy 2- Université Paul Sabatier - IMT - Toulouse

Abstract

We describe the relationship between the notion of directed graph emulator and automata simplification via Myhill-Nerode equivalence. Both being strongly related, we get a new perspective on the genus of regular languages.

In [BD18], we defined the genus of a regular language L to be the minimal genus of the underlying graph of a deterministic automaton recognizing the language L. Technically, given a finite state automaton $\mathtt{A}$, let $L(\mathtt{A})$ denote the language recognized by $\mathtt{A}$ and let $\mathrm{G}_{\mathtt{A}}$ denote the underlying simple directed graph. The *genus* $g(L)$ of a regular language L is the minimal genus of a deterministic automaton recognizing the language:

$$g(L) = \min \{g(\mathrm{G}_{\mathtt{A}}) \mid L(\mathtt{A}) = L, \mathtt{A} \text{ deterministic}\}.$$

We proved that this defines a proper hierarchy of languages. It is clear that the problem is two fold. It is both a question of automaton and a question about graphs. The main topic of this talk is to delineate aspects coming from graph theory to those from automata/language theory and to see the relationships between them. The question is all the more motivated by the observation by Book and Chandra in [BC76] that any language is represented by a *planar non deterministic* automaton. Seen as graphs, one cannot make the difference between deterministic automata and non deterministic ones. So, the question cannot be entirely reduced to its graph aspect.

The graph-theoretical substance of the relation of the minimal automaton to the genus minimal automaton is the notion of *minimal directed emulator*. A closely related notion is that of *directed cover*. The notion of directed emulator is the natural refinement of the notion of graph emulator, introduced by R. M. Fellows in 1985. In order to study the properties of emulators, he also used the notion of graph cover and conjectured that a connected finite graph has a finite planar emulator if and only if it has a finite planar cover. It took more than twenty years before Y. Rieck and Y. Yamashita found a counterexample [RY10] to the conjecture. For directed graphs, it is much simpler: a directed graph has a directed emulator of genus g if and only if it has a directed cover of genus g.

On the other side, one may interpret directed emulators as a transformation that has the strong flavor of an automaton simplification. Our main contribution is to make that sentence precise. That will lead to our main Theorem:

Theorem 1 *Up to isomorphisms, directed emulators are in bijective correspondence with automatic relations.*

This key theorem will lead us to show in which terms the original problem of determining the genus of a regular language is equivalent to the problem of determining the minimal genus of a directed cover of its minimal automaton (Theorem 2).

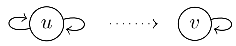
1 Directed emulators

A *digraph* G is a four-tuple (V, E, s, t) with V the set of *vertices*, E its edges *edges* and two maps $s, t : E \rightrightarrows V$ (resp. "source" and "target"). Morphisms between two graphs G and H are pairs of functions (p, q) with $p : V_G \to V_H$, $q : E_G \to E_H$ that preserve source and targets. A graph is simple if there is at most one edge between two given vertices.

Digraphs and digraph morphism between them form a category next denoted $\mathscr{D}$. Simple digraphs and morphisms between them form a full subcategory $\mathscr{D}_S$ of $\mathscr{D}$.

Definition 1 *Let $\phi = (p, q) : G \to H$ be a directed graph morphism. It is a* directed emulator morphism *when: (i) p is surjective and (ii): ϕ verifies the* edge outgoing lifting property*. That is, for any edge $e \in E_H$ and any vertex $x' \in V_G$ such that $p(x') = s_H(e)$, there is an edge $e' \in E_G$ such that $q(e') = e$ and $s_G(e') = x'$.*

The edge e' in clause (ii) is called the emulating edge. We say that a directed emulator morphism is a cover morphism whenever given any edge e and vertex x, there is at most one emulating edge.

Example 1 *Isomorphism are directed covers. A directed emulator shall not be a directed cover as shown by the morphism:*  .

Lemma 1 *Let G' be a directed emulator of a directed graph G. Then there is a directed subgraph G'' of G with the following properties:* (1) *G'' is a directed cover of G and* (2) $V_{G'} = V_{G''}$.

As a consequence, a directed graph has a directed emulator of genus g if and only if it has a directed cover of genus g. We recall that this is not true for undirected graphs (with their respective notions).

1.1 The category of emulators

The composition of two directed emulators (resp. directed covering) morphisms is a directed emulator (resp. directed covering) morphism. The identities being directed emulators (resp. covers), directed graphs and directed emulator morphisms between them form a subcategory Em of $\mathscr{D}$. The category of covers Cov comes with its subcategory Cov_S of covers over simple graphs. None of these inclusion are full.

The categories Em and Cov have no finite limits in general. In particular, there are no equalizers, nor products. However, in a pull back digraph within $\mathscr{D}$, the following holds:

Lemma 2 *Given a morphism $\phi : G \to K$ and a directed emulator $\phi' : H \to K$, in the pull back diagram:*

$$\begin{tikzcd} G \times_K H \arrow[r, "\pi_2|"] \arrow[d, "\pi_1|"'] & H \arrow[d, "\phi'"] \\ G \arrow[r, "\phi"] & K \end{tikzcd}$$

the morphism $\pi_1|$ is a directed emulator. This property does not hold in general for covers.

1.2 An automatic description of directed emulators

In this paragraph we give an alternative description of directed emulators. Let $G = (V, E, s, t)$ be a directed graph.

Definition 2 *Let $\sim_V$ and $\sim_E$ be equivalence relations on respectively the vertices V and the edges E of some directed graph G. The pair $(\sim_V, \sim_E)$ of relations is said to be* automatic *when for all edges e, e', and any vertex x':*

(i) $e \sim_E e' \implies s(e) \sim_V s(e') \;\wedge\; t(e) \sim_V t(e')$;

(ii) $x' \sim_V s(e) \implies \exists e' \in E : e' \sim_E e \wedge s(e') = x'$.

Clause (i) is next called *compatibility* (of $\sim_E$ with respect to $\sim_V$). Clause (ii) is the *bisimilarity* of $\sim_V$ with respect to $\sim_E$. Observe that the definition relates to a graph, not to a morphism.

Given an equivalence relation $\sim_V$ on vertices, let $e \sim_E e' \iff s(e) \sim_V s(e') \wedge t(e) \sim_V t(e')$. Note that $\sim_E$ is compatible with $\sim_V$. If $\sim_V$ is bisimilar with respect to $\sim_E$, the pair $(\sim_V, \sim_E)$ forms an automatic relation. Such a relation is said to be vertex-induced. Think of $\sim_V$ as state equivalence. The definition above is a direct translation of Myhill-Nerode equivalence.

An automatic relation $\sim$ on G induces a new directed graph $G/{\sim} = (V_G/{\sim}, E_G/{\sim}, s/{\sim}, t/{\sim})$ and the canonical directed emulator $G/{\sim} \simeq G$.

Lemma 3 *Any directed emulator morphism $\phi = (p, q) : G \to H$ splits in a unique way as $\phi = \iota \circ [-]_\sim$ where $\sim$ an automatic relation on G and ι is an isomorphism.*

As a consequence, there is a one-one correspondence between emulators and morphisms induced by automatic relations as mentioned in introduction.

The definition of an automatic relation can be refined by introducing labels. A label on a graph (V, G) is a map $\ell : E \to A$ from the set of edges to some finite fixed set. A labelled automatic relation is a pair $(\sim_V, \sim_E)$ such that the compatibility relation (i) is satisfied and such that the *labelled bisimilarity* condition holds:

$$x' \sim_V s(e) \implies \exists e' \in E : e' \sim_E e \wedge \ell(e) = \ell(e') \wedge s(e') = x'.$$

Given a path $c = e_1 \cdots e_n$ in G ($e_i \in E$), let $\ell(c) = \ell(e_1) \cdots \ell(e_n)$. Given a vertex $x \in V$ and a subset W of vertices, let $L_{x,W} = \{\ell(c) \mid \text{all paths } c = e_1 \cdots e_n,\ s(e_1) = x,\ t(e_n) \in W\}$. A labelled automatic relation $(\sim_V, \sim_E)$ is *recursive* if there exists a subset W of vertices such that $x \sim_V y$ if and only if $L_{x,W} = L_{y,W}$. An automatic relation is labelable recursive if there exists a label for which it is recursive.

Theorem 1 *Let G be a directed graph with one accessible vertex. An equivalence relation on G is automatic if and only if it is labelable recursive automatic.*

There is a natural partial order on automatic relations. We define it as follows: $(\sim_v, \sim_E) \leq (\sim'_v, \sim'_e)$ whenever $\sim_v \subseteq \sim'_v$ and $\sim_E \subseteq \sim'_e$. Actually, automatic relations on a graph G form a lattice. Being finite, the lattice has a maximum element. In other words, we can minimize graphs as we minimize automata and the resulting graph is unique (up to isomorphism).

2 The genus of a regular language

Let $G(L)$ be the underlying graph of the minimal automaton recognizing the regular language L. The following holds:

Theorem 2 *Let L be a regular language. Let $n \in \mathbb{N}$. The following assertions are equivalent:* (1) *$g(L) \leq n$;* (2) *the directed graph $G(L)$ has a directed cover G of genus $g(G) \leq n$;* (3) *the directed graph $G(L)$ has a directed emulator G with $g(G) \leq n$.*

Corollary 1 *Any regular language L of size $|L|_{\text{set}} \leq 6$ is planar.*

A directed graph G is *reachable* whenever there is a vertex from which there is a path to any other vertex. The *Langage Genus Problem* is the following problem: given a regular language L and $n \in \mathbb{N}$, the answer is YES if $g(L) \leq n$, otherwise NO. The *Directed Emulation Genus Problem* is: given a directed graph G and $n \in \mathbb{N}$, YES if there is a directed emulator G' of G such that $g(G') \leq n$, otherwise NO.

Corollary 2 *The* Language Genus Problem *has a solution if and only if the* Directed Emulation Genus Problem *restricted to reachable directed graphs has a solution.*

3 Conclusion

We have shown that the Language Genus Problem is decidable if and only if the Directed Emulation Genus Problem is decidable. However, we do not have yet a complete proof of decidability. A general approach consists in properly defining directed minors and proving a "directed graph minor" theorem analogous to the celebrated graph minor theorem of Robertson and Seymour [RS04, §10.5]. This is the approach aimed at in [Kup17]. Even in the case this approach would be successful, one would need to find the minors of nonplanar emulable directed graphs. In some way, we would face the kind of issues that are discussed by M. Chimani, M. Derka, P. Hliněný and M. Klusáček in their article [CDHK13]. Concerning emulators, the relationship between undirected and directed graphs is another promising direction.

References

[BC76] Ronald V. Book and Ashok K. Chandra. Inherently nonplanar automata. *Acta Informatica*, 6:8994, 1976.

[BD18] Guillaume Bonfante and Florian L. Deloup. The genus of regular languages. *Math. Struct. Comput. Sci.*, 28(1):14–44, 2018.

[CDHK13] Markus Chimani, Martin Derka, Petr Hliněný, and Matěj Klusáček. How not to characterize planar-emulable graphs. *Advances in Applied Mathematics*, 50(1):46–68, 2013.

[Kup17] Denis Kuperberg. Directed minors for minimal automata, 2017.

[RS04] Neil Roberston and Paul Seymour. Graph minors. xx. wagner's conjecture. *Journal of Combinatorial Theory, Series B*, 92(2):325–357, 2004.

[RY10] Yo'av Rieck and Yasushi Yamashita. Finite planar emulators for $K_{4,5} - 4K_2$ and $K_{1,2,2,2}$ and Fellows' conjecture. *European Journal of Combinatorics*, 31, 2010.