General complexity bounds for finite dynamical systems

Table of contents

1 Consortium and website

Partner 1 (Marseille)

- Chalopin Jérémie (DR CNRS LIS/DALGO)
- Defrain Oscar (MCF LIS/ACRO)
- Goubault–Larrecq Aliénor (PhD LIS/CANA)
- Guillon Pierre (CR CNRS I2M)
- Monmege Benjamin (MCF LIS/MOVE)
- Ohlmann Pierre (CR CNRS LIS/MOVE)
- Perrot Kévin (MCF LIS/CANA)
- Porreca Antonio E. (MCF LIS/CANA)
- Rolland Marius (PhD LIS/CANA)
- Sené Sylvain (PR LIS/CANA)
- Tapin Léah (PhD LIS/CANA)
- Theyssier Guillaume (CR CNRS I2M)

Partner 2 (Nice+)

- Bridoux Florian (MCF I3S, Nice)
- Durbec Amélia (Postdoc Junia, Lille)
- Gamard Guilhem (MCF Loria, Nancy)
- Richard Adrien (CR CNRS I3S, Nice)
- Riva Sara (MCF CRISTAL, Lille)

Website `https://alarice.lis-lab.fr/`

2 Context, positioning and objectives

2.1 Objectives and research hypothesis

Spirit of the project. One of the most meaningful theorem in computer science is due to Rice, in 1953:

Theorem ([1]). *Any non-trivial question on the behavior of programs is undecidable.*

What is striking about this statement is its generality: it holds for *any non-trivial* question. Furthermore, trivial questions are answered by one of two trivial programs (either returning always "yes", or returning always "no"), hence the dichotomy is both sharp and deep. This brings its interpretation to an epistemological ground regarding the powers and limits of algorithmic problem solving (i.e., by means of a computer, which is nowadays ubiquitous). Attaining this level of generality is the guiding principle of our project ALARICE — to obtain results ***"à la Rice"***.

Rice's theorem belongs to the field of computability theory, and we aim to bring its spirit to the field of complexity theory, which is another fundamental pillar of computer science. This shift towards the decidable world is motivated by the objects under consideration, finite dynamical systems, where virtually any question is decidable via a naive exhaustive search. As a consequence, the formulation of Rice-like results consists in **general (lower and upper) bounds of computational complexity**. The discrete dynamical systems at stake are defined by "simple" local rules, and readily show "complex" global behaviors. They are informally referred to as "complex systems", and we endorse that the computational complexity point of view provides a theoretical framework to formalize this common intuition.

ALARICE is composed of three Objectives. 1) Develop a novel combination of techniques based on the expressiveness of finite model theory, which allows to prove metatheorems, provided a correct definition of non-triviality. 2) Adopt a systematic complexity point of view approach on all facets of automata network theory, aimed at gaining legible knowledge towards the implementation of vaster metatheorem proof techniques. 3) Transfer the results to other models of computation, such as finite cellular automata and reaction systems, through the reconsideration of simulations in the unifying framework of reductions, and reach an understanding of the complexity of finite dynamical systems as abstract and broad as possible.

Mathematical models. The main models we employ are ***automata networks*** (ANs), widespread in the community for their versatility (see the state of the art for applicative aspects). Indeed, any directed graph is the dynamics of some AN, hence any finite (discrete) dynamical system can be modeled as an AN. This is an important feature, ensuring that our developments will be as general as possible. Specific restrictions and transfers to other models of computation are essential aspects of our project. The first purpose of ANs was to model gene regulation processes taking place in the cell nucleus [2, 3].

Formally, an AN is composed of n automata, each holding a state among a finite alphabet Q_i for $i \in [n] = \{1, \dots, n\}$, and a ***local function*** $f_i : X \to Q_i$ giving its next state provided the current global state of the system within $X = \Pi_{i\in[n]} Q_i$. A configuration $x \in X$ gathers the state of all automata, denoted $x_i \in Q_i$ for each $i \in [n]$. We now have a discrete dynamical system (X, f), where $f : X \to X$ is defined as $\forall x \in X : f(x) = (f_1(x), \dots, f_n(x))$ i.e., all automata update their state in parallel at each step. This update policy is also called *fully synchronous*. Other update policies may be observed, such as the *sequential* and *block-sequential* modes updating the automata in a predefined order (repeatedly), the *perfectly asynchronous* mode updating one automaton chosen non-deterministically at each step, and the *non-deterministic* mode where local functions become *local relations* $r_i \subseteq X \times Q_i$ (and y is an image of x when $y_i \in r_i(x)$ for each automaton i). The graph of f on vertex set X is its ***dynamics*** **denoted** $\mathcal{G}_f$. It has out-degree 1 in deterministic settings, i.e., it is a union of cycles with hanging trees pointing towards the cycles. In the theory of dynamical systems applied to the modelling of natural phenomena, the ***limit dynamics*** plays a central role. It is the restriction to configurations of $\Omega_f = \bigcap_{t\in\mathbb{N}} f^t(X)$ where f is applied to sets as $f(Y) = \bigcup_{x\in Y} f(x)$. For deterministic systems, it is a collection of ***fixed points*** $x \in X$ such that $f(x) = x$, and ***limit cycles*** $(x^0, \dots, x^{k-1})$ such that $f(x^i) = x^{i+1 \mod k}$ where k is the length of the limit cycle.

When $Q_1 = \dots = Q_n = \{0, \dots, q-1\}$, we have *$q$-uniform* automata networks, and the case $q = 2$ is the so-called model of *Boolean automata networks* (BANs). We stick on the Boolean case to introduce a central object of AN theory: the ***interaction graph*** **denoted** G_f. It has one vertex per automaton, and an arc from i to j whenever $f_j : \{0,1\}^n \to \{0,1\}$ effectively depends on its i-th component (formally, if there exists $x \in \{0,1\}^n$

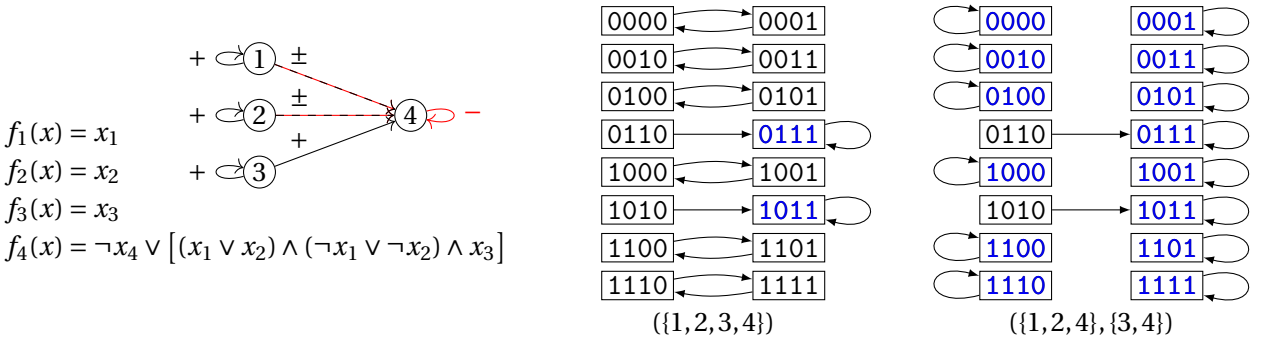

Figure 1: A Boolean automata network of size $n = 4$ and two dynamics. Left: local functions and interaction graph G_f (the arcs from 1 to 4 and from 2 to 4 are both positive and negative). Center: parallel dynamics, it corresponds to updating the full set of automata $\{1,2,3,4\}$ at each step. Right: dynamics obtained when updating first automata $\{1,2,4\}$ then automata $\{3,4\}$. Fixed point configurations are depicted in blue.

such that $f_j(x) \neq f_j(x+e_i)$ with e_i the i-th base vector and addition taken modulo 2). The graph G_f captures the **architecture of the network**, through the mutual influences among automata. Signs may be added to the arcs of G_f, representing two non-exclusive types of influence: positive or negative. The sign of an arc is positive (resp. negative) when there exists a configuration x such that flipping x_i from 0 to 1 flips f_j from 0 to 1 (resp. from 1 to 0). Intuitively, along a positive arc the head tends to mimic the tail, whereas along a negative arc the head tends to negate the tail. An example is given on Figure 1.

Given our focus on algorithmic complexity theory, it is important to detail how instances of our model are encoded. In line with the encoding of propositional formulas in the Boolean case, an AN of size n is given as a list of n Boolean **circuits**, one for each local function f_i with $\sum_{j\in[n]} \lceil \log_2(|Q_j|) \rceil$ input bits and $\lceil \log_2(|Q_i|) \rceil$ output bits. To avoid a costly (coNP-complete) instance validity check, outputs are interpreted modulo $|Q_i|$. In case of a non-deterministic local relation r_i, the circuit has $\sum_{j\in[n]} \lceil \log_2(|Q_j|) \rceil + \lceil \log_2(|Q_i|) \rceil$ input bits and 1 output bit. Circuits provide a malleable way to express local behaviors. Any directed graph $G = (V, A)$ is the dynamics of an AN of size 1 and alphabet V. The obtained circuit corresponds to the notion of ***succinct graph representation***, which will be discussed in Objective 1. In case of bounded in-degree of the interaction graph $\Delta(G_f) \leq d$, which is meaningful both in theory and practice, a truth-table encoding of local functions can take polynomial space $\mathcal{O}(n^{d+1})$. In this case the notion of communication graph is necessary, which will be discussed in connexion with Objective 2.b. Lastly, in Objective 2.a. we consider as input the interaction graph G_f, encoded by its signed adjacency matrix.

The plasticity of automata networks, and its close connection with circuit representations, is able to give some immediate complexity lower bounds. For example, the BAN from Figure 1 in parallel proves the NP-hardness of the fixed point existence problem, by reduction from SAT (here with $\varphi = [(x_1 \vee x_2) \wedge (\neg x_1 \vee \neg x_2) \wedge x_3]$ on 3 Boolean variables). Additional constraints, such as a bounded degree of interactions, the monotonicity of local functions, or fixed/uniform/bounded alphabets, can strengthen the considerations and help investigating the core attributes of the model that are able to embed the behavioral complexity. They are also steps towards **other models of computation**, such as reaction networks (corresponding to simple forms of local functions) and finite cellular automata (where the spatial uniformity imposes a rigid architecture on the network). These interconnections will be investigated in Objective 3.

Objective 1 – Unveil metatheorems. Standard dynamical approaches aim at understanding the dynamics $\mathcal{G}_f$, in terms of the local functions $(f_i)_{i\in[n]}$. This follows methodological reductionism, a philosophy widely adopted in theoretical and applied sciences such as system biology (see the state of the art section for references). This leads to algorithmically solving decision problems of the form: given $(f_i)_{i\in[n]}$, does $\mathcal{G}_f$ has a given property? Such questions are able to capture in a formal way the paradigm that complex behaviors can emerge from simple interactions (e.g. there are PSPACE-complete problems for ANs with local functions of in-degree 2). Many problems are known to be computationally complex in this framework (a partial list is given in the state of the art section), from NP to PSPACE (even NEXPTIME, but PSPACE is arguably a natural upper bound for many models, because the whole dynamics can be inspected in polynomial space,

although without a full instantaneous memory of the visited configurations).

With the circuit encoding presented above for ANs, capturing a common way of expressing the mechanics of local behaviors (hence the terminology *automata* for the entities), many problems can easily be proven to be NP-hard or coNP-hard. In 1985, Alon stated the NP-completeness of the fixed point existence problem by saying "we omit the details" [4]. Carrying on our comment of Figure 1, this construction also proves the coNP-hardness of asking whether the parallel dynamics is a union of limit cycles of length 2 (by reduction from UNSAT), and also of testing the injectivity of $\mathcal{G}_f$. Its structure can also be used to prove that the existence of a limit cycle of some fixed length is NP-complete [5], as well as the existence of other induced subgraphs fixed in the problem definition. The idea is to have $n+m$ automata, such that a configuration evaluates some propositional formula on n Boolean automata, and accordingly creates one of two subgraphs in $\mathcal{G}_f$ on the corresponding set of configurations where only the last m automata vary (for the parallel example of Figure 1, either a cycle of two configurations if the valuation is negative, or a fixed point with a pending node if the valuation is positive). It is possible to use analogous constructions to obtain complexity lower bounds for other properties of $\mathcal{G}_f$ [6]. The property of being a constant map is also coNP-complete, but requires a different construction (to reduce from UNSAT), because configurations corresponding to negative valuations are all sent to the *same* configuration (for example with all automata in state 0). Building on this analysis, we have recently obtained a first metatheorem encompassing at once a large range of such developments, using tools from finite model theory:

Theorem 1 ([7])**.** *Given a deterministic AN f encoded as Boolean circuits, **any non-trivial question** on $\mathcal{G}_f$ expressed in **first-order graph logic** is* NP*-hard or* coNP*-hard.*

The choice of the language of model theory allows to express questions with a high level of abstraction and generality. A crucial point in this context is to craft a good notion of non-triviality. In this case, a question is non-trivial if it admits both infinitely many models (local functions of ANs verifying the property), and infinitely many countermodels (local functions of ANs not verifying the property). Consequently, trivial questions are answered in constant time, simply by checking among a finite list of instances. This metatheorem fully dichotomizes the computational complexity of deterministic ANs with respect to first-order questions, showing a meaningful and sharp gap, in the spirit of Rice, between $\mathcal{O}(1)$ and NP- or coNP-hardness. **This is the kind of Rice-like theorem we aim at developing**.

The technicalities are numerous. Let us first recall that first-order graph logic (FO) are formulas evaluated on graphs (finite graphs in our setting, namely $\mathcal{G}_f$), constructed with the usual connectives $\wedge, \vee, \neg, \Rightarrow$ and quantifications $\exists, \forall$ on vertices (configurations in our setting). Atoms are built on the signature $\{=, \rightarrow\}$ where $x \rightarrow y$ is true when there is an arc from x to y. To a FO formula we associate a decision problem, taking as input the circuits $(f_i)_{i\in[n]}$. It includes all the examples mentioned above, and many more:

- existence of a fixed point: $\exists x : x \rightarrow x$;
- existence of a limit cycle of length three: $\exists x, y, z : (x \rightarrow y) \wedge (y \rightarrow z) \wedge (z \rightarrow x) \wedge (x \neq y) \wedge (x \neq z) \wedge (y \neq z)$;
- injectivity: $\forall x, x', y, y' : (x \rightarrow y) \wedge (x' \rightarrow y') \wedge (y = y') \Rightarrow (x = x')$;
- being constant: $\forall x, x', y, y' : (x \rightarrow y) \wedge (x' \rightarrow y') \Rightarrow (y = y')$;
- being the identity: $\forall x : x \rightarrow x$;
- having exactly one fixed point: $\exists x : (x \rightarrow x) \wedge (\forall y : (y \neq x) \Rightarrow \neg(y \rightarrow y))$;
- having at least three configurations (a trivial property): $\exists x, y, z : (x \neq y) \wedge (x \neq z) \wedge (y \neq z)$;
- being a map (a trivial property): $\forall x \exists y : x \rightarrow y$;
- ...

One can readily notice that some of these problems are NP-complete whereas others are coNP-complete (indeed, formulas can be negated). Thus, unless a long-standing open complexity collapse is proven, a general complexity lower bound must accommodate this **symmetry regarding negation**, as Theorem 1 does. Furthermore, different formulas lead to widely different behaviours requiring a range of techniques (see the three different gluing operations in [7]).

We aim at developing complexity metatheorems, on three levels:

- from the local functions of ANs to their dynamics for Objective 1,
- towards other AN-related problems for Objective 2,
- towards other models of computation for Objective 3.

The proof techniques employed to obtain the metareduction in [7] are novel, and required to combine tools from finite model theory (in particular Ehrenfeucht-Fraïssé games and Hanf-Gaifman locality) with the constructions of discrete dynamical systems. **It can be seen as an abstract pumping** corresponding to the problem SAT/UNSAT we reduce from: negative valuations generate neutral graph pieces to the dynamics (here is the pumping), and positive valuations generate pieces that make a difference (see below the saturation graph). This suggests a number of extensions, none of which is evident. Therefore we have a collection of concrete questions to address at short/mid-term:

- **Can we obtain an analogous result for ANs on bounded alphabets** (e.g. Boolean)**?** At present, [7, Theorem 5.2] deals with succinct graph representations rather than simply ANs, because unbounded alphabets are required. Indeed, the essence of AN theory lies in the interactions, and we want to go beyond the graph theoretical field opened by Galperin in 1982 [8]. This amounts to using finite model theory more finely, in order to construct dynamics with a prescribed number of configurations (power of the alphabet size). We have early arithmetical evidence that the Boolean case may be simpler to obtain than arbitrary q-uniform networks, because a geometrical sequence can be extracted from $\{ak+b \mid k \in \mathbb{N}\} \cap \{2^n \mid n \in \mathbb{N}\}$, namely the sizes of dynamics being both obtained by pumping (left term, for a base graph of size b and a pumped part of size a), and decomposable as an integer number of Boolean automata (n). The use of Fermat's little theorem, and the gluing of graph parts beyond deterministic dynamics, are still to be precised. Nevertheless, this is our most desired objective, in order to **bring metatheorems to the AN community at large**.
- **What about other signatures?** Observe that the signature $\{=, \rightarrow\}$ is up-to-isomorphism (it ignores configuration labels, i.e., automata states). Including finer relations would be meaningful for applications, and this direction is largely open. If we add relations to distinguish configurations, such as the componentwise partial order $\leq$ or the total order, then we have P-complete problems, such as $\forall x : (\forall y : x \leq y) \Rightarrow (x \rightarrow x)$ meaning that the minimum configuration (0^n in the Boolean case) is a fixed point (checked simply by evaluating the local functions). Hence a general complexity lower bound would be a P-hardness result, and **logspace metareductions** are required. This extension is meaningful for applications, where different configurations may bear different interpretations. Goubault–Larrecq did her Master thesis on the matter in 2022, and found how to: 1) meticulously design circuit constructions in logarithmic space for the reductions from [7], and 2) prove a Rice-like P-hardness lower bound when unary relations distinguishing unique configurations (one per network size) are added. This is encouraging, but leaves room to push the subject forward.
- **Can we extend the study to non-deterministic ANs?** In this setting, new non-trivial FO questions arise, which must be proven hard as well. Deterministic ANs correspond to $\mathcal{G}_f$ having out-degree 1, a hypothesis thoroughly exploited in the proofs of [7]. In a preprint [9] (also bringing the study to MSO, see next question), we have shown that succinct representations of general digraphs can have model sizes such that the design of a polytime reduction requires almost a P = NP collapse (on a *robust set of instances*; even in FO). However, we have also found that **bounding a structural parameter of the dynamics** $\mathcal{G}_f$ is a key tool to go beyond this lock. We have obtained a complexity lower bound metatheorem for the NP- or coNP-hardness of problems expressed by **arborescent** formulas, i.e. those having both infinitely many models of bounded treewidth, and infinitely many countermodels of bounded treewidth. The tree structure of decompositions then allows to extract, with the aid of the finite number of equivalence classes for its nodes (basic model theory application), pieces to be glued to win Ehrenfeucht-Fraïssé games. That is, *to pump*. We want to try other graph families with finitary constraints on the dynamics $\mathcal{G}_f$, and formalize in a larger setting what are the "**pumpable graph parameters**". We expect that bounded cliquewidth works, which would include some problems not captured by bounded threewidth, for example the property of being a clique. Note that we do not parameterize the input $(f_i)_{i\in[n]}$, but the dynamics $\mathcal{G}_f$. Individually, such results give sufficient conditions on the formulas (restrictions on the notion of non-triviality) to obtain a general complexity lower bound. Collectively, we believe that they are building up our understanding of what we still informally call "the pumping technique".
- **And to monadic second-order (MSO) logic?** MSO is more expressive, but the tools employed in [7] for FO (in particular Hanf-locality) do not apply any more to quantifications over sets of vertices. The new constructions from [9] manage to perform abstract MSO pumping, hence overtaking FO's local-

ity. AN complexity theory may be **a new applicative field for finite model theory**. For example, it is proven in [9] that, for any $m \in \mathbb{N}$, there exists a graph Ω_m saturating all MSO formulas of quantifier rank m. That is, for every formula ψ with rank m, $\Omega_m \cup G \models \psi \iff \Omega_m \models \psi$. For each formula, in order to achieve a reduction, it remains to exhibit a pumpable part in the graph, as discussed above. We expect to foster the interactions with tools from finite model theory, in particular by considering metareductions from problems with a different shape/structure than SAT. This will also arise when trying to characterize formulas expressing higher complexity classes: [7] also proves that, in the deterministic setting, FO achieves completeness for all levels of the polynomial hierarchy, hence the general complexity lower bound can be refined. Typical MSO questions (unexpressible in FO) such as connectivity are PSPACE-complete [10], therefore we also target general complexity bounds for this class, which is central in the realm of universality of computations operated by discrete dynamical systems.

Objective 2 – Exhaust typical problems. Before one can imagine Rice-like metatheorems, it is necessary to get a vision of typical complexity results that hold for large classes of questions, and the AN constructions at stake. Therefore, we also aim at developing other topics in AN theory (other than the central questions from local functions $(f_i)_{i\in[n]}$ to dynamics $\mathcal{G}_f$, currently in the scope of Objective 1) with a systematic computational complexity point of view. Constructions often rely on structural characterizations (in terms of network architecture, as it is captured by the interaction graph G_f), hence subobjectives will also tackle non-algorithmical issues, still in the spirit of our project.

Subobjective 2.a – On input G_f. When only the signed interaction graph G_f of a Boolean network is known, many local functions are possible: a **doubly exponential number**, in general. Constructing a compatible BAN is not always possible (for example it is impossible that an automaton have in-degree 1 and an arc signed $\pm$), but the admissible interaction graphs are known [11]. In the **monotonous** setting, that is when all Boolean local functions are nondecreasing for the bitwise partial order on configurations ($x \geq y \implies f_i(x) \geq f_i(y)$), possible automata networks are a product of Dedekind numbers (for which finding an explicit formula has been open for 125 years). They are tied up with the notion of Boolean lattice dualization, to which every monotonous local function is a solution (deciding whether all solutions have been obtained in that context is one of the few natural problems known to be in QP but not in P [12]). Deriving knowledge on possible dynamics $\mathcal{G}_f$ solely from the knowledge of the interaction graph G_f is therefore a challenging task, and a central topic of AN theory. It is also especially meaningful towards the applicative field of system biology and in particular gene regulation, where **wet-lab experiments** mostly concentrate on discovering the architecture of the network (the arcs of G_f and their signs) [13]. Hence, such problems are at the core of applications.

The complexity of questions on the maximum and minimum possible number of fixed points have been settled in [11], ranging from P (with a tough algorithm from [14]) to NP and up until NEXPTIME (for example: given G, does there exist f with no fixed point and $G_f = G$?). This series of results somewhat **explains why structural bounds cannot be tight** (known bounds are sometimes very loose, such as the exponential upper bound in the positive feedback vertex set of G_f [15]). It is notable, though expected, that the complexities are lowered when the input interaction graphs have bounded degree (completeness for Σ_2^{P} and $\mathsf{NP}^{\#\mathsf{P}}$). **What about unsigned interaction graphs, and other quantities such as limit cycles?** The tricky constructions from [11] exploit known structural properties of fixed points presented in [15, 16], and **the current lack of knowledge on limit cycles needs to be addressed first**. The study of **non-Boolean alphabets** requires beforehand to adapt signs with the increased degree of liberty (when only G_f is known).

Subobjective 2.b – Compute G_f. Given the Boolean circuits encoding the local functions of a BAN f, it turns out that it is not obvious to compute its interaction graph G_f (see [6, Chapter 4]). In general the associated decision problem is complete for the class DP, which is sandwiched within $\mathsf{NP}, \mathsf{coNP} \subseteq \mathsf{DP} \subseteq \Sigma_2^{\mathsf{P}}$. However, **some properties are easier to check than others**. For example, deciding whether G_f is a clique is complete for NP, deciding whether G_f is empty is complete for coNP, and G_f can be computed in polytime under the promise that it has bounded in-degree. In light of these early observations, **we aim at a tetrachotomy theorem** into these four classes, to fully characterize the complexity of problems related to computing the properties of G_f. Sufficient conditions for the lower bounds can be derived from known constructions, thus the difficulty in this approach is to find an appropriate **general notion of graphical**

property to allow this classification.

These lower bounds (in particular the DP-hardness of computing G_f in general) also mean that algorithms considering as input both the local functions and the precise interaction graph are implicitly assuming a costly promise. Hence the concept of **communication graph** currently circulating in the community is especially useful to replace the interaction graph. It corresponds to a superset of the interactions, i.e., an automaton depends on *some* in-neighbors, but may ignore some others (this can be interpreted as a syntactical check that only its in-neighbors appear in the definition of its local function). This also strengthens the interest for families of networks defined by their interaction graph G_f alone, such as disjunctive, totalistic and set-defined networks [17]. Such families permit to bypass some complexity locks.

Subobjective 2.c – Knowledge of $\mathcal{G}_f$ up to isomorphism, or partial. In [18] a fairly new question is raised, taking the counterpart to standard approaches: what if the dynamics $\mathcal{G}_f$ is known up to isomorphism, but the network itself is unknown? Without the isomorphism condition, the local functions are completely determined by $\mathcal{G}_f$, hence a more meaningful question concerns the possibility of extending a partial knowledge of $\mathcal{G}_f$ (see below). In the Boolean setting, a dynamics $\mathcal{G}$ up to isomorphism may admit various BANs f, with various interaction graphs G_f (without signs, as a first exploration). Therefore one asks about the set $G(\mathcal{G}) = \{G_f \mid \mathcal{G}_f \sim \mathcal{G}\}$ **for some** $\mathcal{G}$. Insightful questions on the number and degrees of possible interaction graphs are answered in [18] in the deterministic parallel setting, notably leading to a strong universality of the complete interaction graph: when $\mathcal{G}$ is neither the identity nor a constant map (each having a unique element in $G(\mathcal{G})$), then $G(\mathcal{G})$ contains the complete interaction graph. In other words, the complete interaction graph can support all dynamics except those two. **Structural bounds on the constraints that can be imposed on** $G(\mathcal{G})$ (e.g. finding some $\mathcal{G}$ that maximizes the minimum in-degree or number of arcs in its interaction graphs) still need to be tighten. We also **conjecture that there are univeral dynamics** having almost all possible interaction graphs, i.e. that $\max\{|G(\mathcal{G})| \mid V(\mathcal{G}) = \{0,1\}^n\} \sim_{n\to\infty} 2^{n^2}$. Trying to minimize $G(\mathcal{G})$ is a complementary task. We also aim at understanding which families of dynamics $\mathcal{G}$ can be supported by an interaction graph with bounded maximum in-degree. Such dynamics can then be generated using only finitely many local functions. In that sense, local computations are simple, but the resulting dynamics are still very difficult to predict. A basic counting argument shows that almost no dynamics are degree-bounded, but rather surprisingly it is difficult to exhibit non-trivial families. For instance, we do not know whether the dynamics $\mathcal{G}$ consisting of a single limit cycle (of length 2^n in the binary case) is degree-bounded or not [19].

Another concrete open question is, for example, to **characterize the set** $\mathcal{G}$ **such that** $G(\mathcal{G})$ **contains an acyclic interaction graph**. According to a fundamental observation by Robert [20], if G_f is acyclic then f^n is a constant map (i.e. f converges to a unique fixed point in at most n steps). This basic result says that **cycles of interactions** (*feedback loops* in a biological context) **are the engines of behavioral complexity**. This inverse problem revives this relationship, which has not been investigated since the 1980s. The characterization is not obvious, and may be analyzed as the possibility of "folding" $\mathcal{G}_f$ in at most n step.

A partial knowledge of $\mathcal{G}_f$ (which is another view of experimental outcomes in applicative context) can always be extended to a full AN, because any directed graph is the dynamics of some AN. A pertinent additional constraint is **monotonicity**, which may be imposed to *all* or *some* local functions. In the fully monotonous Boolean setting, searching **an extension corresponds to the famous lattice dualization** problem [12] (with a $N^{o(\log N)}$ algorithm, when N is the size of the known domain). Indeed, the problem can be split to the level of the extension of individual local functions. Partially non-monotonous extensions opens the field, and **good complexity upper bounds** are also meaningful for practical purposes. Moving to counting the number of solutions is also a means to guide experiments minimizing the uncertainties.

Subobjective 2.d – Update modes. The consideration of other update policies opens a considerable world to explore. **Virtually any of the preceding problem may be revisited**, for example by asking whether there exists an update schedule within a prescribed family. In terms of computational complexity, quantifying over this new parameter may bring problems one level higher in the polynomial hierarchy, as shown in a first study of limit cycles under block-sequential update modes [5]. This can be interpreted as **the update schedule itself embedding part of the complexity**. Some applicative outlooks may guide us throughout (although the most pertinent update policies are still largely discussed in gene regulation for example [21]), and we expect to be able to overtake usual update modes. The asynchronous and all block-sequential up-

dates are known to preserve the set of fixed points (the canonical initial focus on the limit dynamics), but the recent introduction of **block-parallel update modes** (dual to the latter) **break this invariance**. They also permit a broader embedding of complexity in the dynamics of substeps (in the sequence of blocks sequentially updated to compute one step), basically smashing all problems to the class PSPACE, with notable exceptions: for example testing bijectivity is still coNP-complete in this context [22]. These early results with new constructions need to be developed, to reach the level of metatheorems (only sketched so far).

The **interplay** between elements of flexibility offered by update schedules, and structural constraints on the interactions or the local functions, is also expected to increase our understanding of the influence of update modes. For example, [23] shows that asynchronism can compensate the limitations coming from symmetric local interactions. The results are expressed in terms of computational complexity, but also in connexion with simulations and Objective 3. Given that this topic brings numerous new questions, restrictions to monotonous ANs with G_f of in-degree 1 or 2 will be a first step in this exploration.

Objective 3 – Go beyond automata networks. ANs are flexible enough to model any finite dynamical system, which is a strength of the model in terms of universality. However, it can also be a weakness, in the sense that without further restrictions ANs are "too versatile" and do not model distributed processes that are decomposable. As an illustration, having a single automaton ($n = 1$) allows to have arbitrarily many vertices in the dynamics (whose vertex set is the configuration space), but it corresponds to *succinct graph representations*, therefore missing the flavor of natural phenomena where the global behavior emerges from local interactions among entities. Our project is to go beyond this pitfall both ways: 1) by considering additional restrictions on its architecture (G_f) and local functions ($(f_i)_{i\in[n]}$), in order to transfer the complexity bounds to other models of computation via **simulations acting as reductions**, and 2) by **fully abstracting the model of computation**, hence see the dynamics as pure directed graphs and consider general operations for their decomposition. We aim to **demonstrate that our studies are not trapped to the model of ANs, but indeed more general**.

Subobjective 3.a – Simulation. The complexity point of view on finite dynamical systems is widespread, with problems characterized by completeness results on various models such as finite cellular automata [24], reaction systems [25], tile assembly [26], DNA folding [27]. Our goal is to **lift the complexity bounds to the level of metatheorems** as in Objective 1. To this purpose we aim at **developing strong formal simulations, allowing to transfer complexity bounds among models of natural computation**. Concretely, a target is to formalize statements of the form: **if** A **simulates** B **via a transformation** $\phi : X_B \to X_A$ **then the complexity of** B **cannot exceed that of** $A + \phi$. The difficulty lies in finding satisfactory definitions i.e., as general as possible. A simulation ϕ is roughly expected to be a (possibly partial) factor map, that is with $\phi \circ B = A^k \circ \phi$ (i.e. k steps of A simulate one step of B). In order to gain pertinence and sharpness for the purpose of complexity bounds, ϕ should be as simple as possible. Constraints on ϕ can be formalized in multiple ways, each of them dedicated to meet particular models of computation. A subset of our current landscape is: strictly step-by-step ($k = 1$), non-uniformly step-by-step (k varies), total (ϕ surjective), bijective (so is ϕ), and asymptotic (on the limit dynamics). These restrictions and relaxations can also be combined. Broadly speaking, our group is interested in developing an **abstract theory of simulation**, aimed at catching our intuition towards what we would call a **simulation thesis**.

Back to ANs, to unlock the transfer of current metatheorems [7, 9], a major step is to bring it to **bounded alphabets** (e.g. Boolean, which is part of Objective 1). This would allow a decomposition into automata performing local computations, to be matched with the corresponding entities of other discrete dynamical systems. As an illustration, in order to reach finite cellular automata (i.e. having a simple ϕ mapping ANs to CAs), another identified stage is to bring uniformity to the local rules. Here, attainable **restrictions on the interaction graph of ANs constructed with the pumping technique** will be crucial, and it is also expected to be nourished by our progresses on sub-dynamics (directed graphs) obtained from model theory (so far, only basic tools are employed). A simple ϕ would readily act as a reduction, provided that our ANs are sufficiently constrained. Reaction networks are closer to ANs, and the transfer may be more direct (obtaining disjunctive networks with signs goes in the same direction). An interesting case of study will be P-systems from membrane computing, because their relation to complexity theory is unconventional, such as SAT being famously solvable in deterministic polytime using active membranes [28]. Obtaining **simulation impossibilities** is a graal, but complexity theory assumptions can help to circumvent the current lack

of a satisfying notion of "not being Turing universal" (despite considerable interest).

We are used to see SAT and Cook-Levin theorem as the foundation of complexity theory, but canonical complete problems are trivially expressed in terms of Turing machine simulation (typically for a given number of time steps, see for example [29, Proposition 3-M]). It means that **complexity lower bounds implicitly give efficient ways to embed arbitrary computation**. As a broader objective, we want to **explicit how** (possibly non-deterministic) computations are operated in hardness proofs, because reductions *are* simulations and we *do* embed algorithms in many facets of our discrete dynamical systems.

Subobjective 3.b – Algebra of finite dynamical systems. Decomposition methods are widely used to decrease the cost of extensive computation, in a divide-and-conquer approach [30, 31, 32]. Two graph operations are especially relevant to dynamical system theory: 1) alternative composition (*one or the other*, i.e. disjoint union of dynamics) denoted +, and 2) simultaneous composition (*both*, i.e. the direct product of dynamics) denoted $\cdot$. **Fully abstracting the computational model**, those two operators on concrete dynamics (directed graphs as adjacency matrices, taken up to isomorphism) have recently been shown to form a semiring [33]. This provides a framework for the analysis of complex behaviours that are actually due to smaller independent systems. Landmark elements of $\mathscr{G}\cdot\mathscr{H}$ are **decomposable**, such as fixed points (they must exist in both $\mathscr{G}$ and $\mathscr{H}$) and transient lengths (maximum path lengths from $\mathscr{G}$ and $\mathscr{H}$). The (direct) product of two dynamics can also be interpreted in terms of succinct representations as the disjoint union of their two circuits, and we want to further explore these connections.

Our goal is to characterize the complexity of polynomial equations of the form $P(X_1,\dots,X_n) = Q(X_1,\dots,X_n)$ (there is no notion of subtraction, hence the placement of the terms matters). Such equations are undecidable in general (a consequence of Matiyasevich's theorem), therefore we aim at understanding decidable families of polynomial equations, such as those with a constant right-hand side. Members of the consortium are currently focusing on $\mathscr{H}\cdot X = \mathscr{G}$, that is the **problem of divisibility** (of $\mathscr{G}$ by $\mathscr{H}$), which is unexpectedly subtle [34]. Indeed, part of the difficulty lies in the non-injectivity of polynomials, because a decomposable dynamics possibly admits **multiple factorizations** (bounding their quantity is also a target) [35]. This is sustained by our **recent discoveries** that computing roots (solutions of $X^k = \mathscr{G}$) can be done in cubic time because of the injectivity of the polynomial X^k, and that general linear multivariate equations (of the form $\sum_{i\in[n]} \mathscr{H}_i \cdot X_i = \mathscr{G}$) are NP-complete (writing in progress). It is also not yet known **whether prime dynamics** (basic building blocks) **exists**. **The field is new and largely open**.

2.2 Position of the project as it relates to the state of the art

We follow the structure of its objectives to present the originality of ALARICE, and our contributions.

Objective 1 – Unveil metatheorems. **The use of model theory to state metatheorems is entirely new**, and has been developed by four members of the consortium in a recent paper [7] for FO graph logics in deterministic setting with unbounded alphabets (Theorem 1). We have a preprint to carry the results to MSO with non-deterministic dynamics [9], with an additional condition on the non-triviality of formulas, expressed in terms of structural graph theory (arborescence). It is also proven in [9] that an unparameterized version of triviality fails (under complexity assumptions developed in [36]), and we aim to generalize our result under pumpable structural constraints other than treewidth. We have preliminary results on q-uniform ANs for $q = 2$, and are willing to study it directly to the level of MSO, and for both deterministic and non-deterministic dynamics (each with original constructs, gluing delicately inside tree decompositions). The exposure of [7] furthermore includes complementary general PSPACE-completeness bounds for abstract properties of limit sets and their sizes (remark that limit dynamics drop alphabet issues). Rice-like complexity statements encompass at once many results from the literature: [4, 37, 38, 39, 5] (see [6] for a survey). In non-deterministic settings, that is with arbitrary directed graphs, formal developments are more abstract than for deterministic cases, where one can exploit the simple structure of out-degree-1 dynamics. It therefore needs finer tools from structural graph and finite model theories, for which the consortium gathers extra expertise [40, 41, 42, 43, 44, 45].

At present, the results from [7, 9] belong to the field of **succinct graph representations** (mostly regarding general non-deterministic dynamics), because of an issue in obtaining the correct number of nodes in the Cartesian product $V(\mathscr{G}_f) = X$. Metareductions (for upper and lower bounds) are also central in the literature

of succinct graph representations, to transfer known complexity bounds on graph problems with standard encodings (such as adjacency matrix): 1) in [46] from P (non-succinct) to NP (succinct), such as existence of a triangle in $\mathcal{G}_f$, or maximum in-degree $\Delta(\mathcal{G}_f) \geq k$ for some fixed k. They also obtain Σ_2^{P}-completeness for minimum in-degree $\delta(\mathcal{G}_f) \leq k$ for some fixed k, and leave many loose bounds, 2) in [47] from NP-hard (non succinct, for projection reduction) to NEXPTIME-hard (succinct), such as whether $\mathcal{G}_f$ has a Hamiltonian cycle, or whether it is 3-colorable, 3) in [10] the authors clarify the requirements on reductions to be convertible to succinct encodings, and give a nice class to class general conversion (they also study problems in the counting hierarchies), and 4) in [48] lower complexity classes are considered (basic search problems becomes PSPACE-hard). Our approach **differs**: we do not transfer known results, but establish new holistic complexity lower bounds from the expressiveness of graph logics. This literature however provides good material regarding the feasible refinement of metatheorems to other complexity classes, because it characterizes **many well-studied graph problems** for succinct encodings.

Interestingly, our Rice-like complexity lower bound for MSO on dynamics with bounded treewidth [9] can be seen as a **succinct counterpoint to Courcelle's theorem** [49]. The proof technique based on graph gluing (pumping) closely relates to the notion of *composability* in model theory [50, 51]. The composition method literature focuses on the decidability of theories and, problematically to our purpose, abstract operations often distort the signature of the logics. Nevertheless, our pumpings on unbounded-depth trees for deterministic ANs, and on tree decompositions for non-deterministic ANs, are similar to **Muchnik's *iterations*** of a structure [52, 53], and we want to investigate this connection.

In a larger spectrum, obtaining general results in the spirit of Rice's theorem has been pursued in various ways. Notably in the field of **cellular automata**, with **successful undecidability** results on their limit sets [54, 55, 56]. In the realm of complexity theory, there are attempts towards Rice-like theorems for non-trivial counting property of circuits, but they get stuck to UP and SPP hardness [57, 58]. This track has been completed in [59], proving that any language property of NP-sets is NP-hard to decide (the same holds for other leaf languages). Given that Rice's theorem proves their undecidability, the significance is limited. A more fruitful approach is di/tri/tetrachotomy theorems, discussed in Objective 2.b.

Objective 2 – Exhaust typical problems. The multiple facets of AN theory brings a variety of problems to address, connecting the three objects: local functions $(f_i)_{i\in[n]}$, interaction graph G_f and dynamics $\mathcal{G}_f$. They serve the qualitative modelling of natural phenomena with a wide range of applications: from biology to sociology and economics [2, 3, 60, 61, 62, 63, 64]. In some contexts **monotonicity** is requested, and it is also a case of study. From a practical point of view, computational complexity gives bounds essential to the design of (the most) efficient algorithms. From a theoretical point of view, it gives formal meanings to the intuitive term "complex system". **Classical results of the domain are structural**. For example [20, 65, 15, 66, 67, 68] relate properties of G_f such as the size of feedback arc sets, to properties of $\mathcal{G}_f$ such as its number of fixed points. In particular, Thomas's rules are easily catchable statements: 1) having at least two fixed points in $\mathcal{G}_f$ requires a cycle with an even number of $-$ arcs in G_f, and 2) having no fixed point requires a cycle with an odd number of $-$ arsc in G_f (see [69, 70] for surveys). Our proposal is to understand these relations through the lens of computational complexity theory. **This renewed point of view outlines why some structural bounds are still loose despite considerable efforts**.

Subobjective 2.a – On input G_f. A primary focus on $\mathcal{G}_f$ are its **fixed points**, which can be interpreted as cellular types on gene regulatory networks [2, 3, 60]. It has been shown that the number of fixed points is at most exponential in the minimum (positive) feedback vertex set of G_f, and is at least its packing number [71, 15, 72]. **These bounds are tight, but sometimes far from optimal**. The question has been raised, from a computational complexity perspective, by Richard during GDR-IM days in 2016. Four members of our consortium have obtained the first results after intensive efforts, combining a new kind of construction (for the reductions) with known structural relations [11]. Computing the maximum and minimum number of fixed points possible on a given G_f touches a large panel of complexity classes, in particular under bounded degree of interactions ($\Delta(G_f) \leq d$, with $\mathsf{NP}^{\#\mathsf{P}}$-completeness), and also required the use of a strong graph algorithm to detect positive cycles (with an even number of $-$ signs) [14]. These are the **first results of this kind**. They are particularly **insightful for applications** (where G_f is well approximated, but $\mathcal{G}_f$ unknown) and softwares [73], because our lower bounds say that the problem is readily intractable (except for asking whether one fixed point is possible, but to the best of our knowledge, the algorithm from [14] has not

yet been successfully implemented). In order to develop it, a deep understanding of structural relations is essential, but for example on limit cycles, it is currently limited to Thomas's rule [66, 69] and the characterization of very simple architectures [67]. High and varied complexities **point out the obstacle to the design of good structural bounds**.

Subobjective 2.b – Compute G_f. This topic only appears in the coordinator's HDR thesis [6], and sheds light on the cost of implicit promises when both the local functions $(f_i)_{i\in[n]}$ and the interaction graph G_f are given as input to an algorithm. It is related to the notion of **essential variable** in Boolean functions [74], whose complexity was not fully characterized by DP. The **known range** from P (bounded degree) to NP (arc existence only), coNP (arc inexistence only) and DP (in general) hint at the quest for a tetrachotomy theorem to classify all graph properties. Besides Schaefer and Creignou-Heramnn famous dichotomy theorems, there are trichotomy and tetrachotomy results in the literature [75, 76, 77].

Subobjective 2.c – Knowledge of $\mathcal{G}_f$ up to isomorphism, or partial. The systematic study of $G(\mathcal{G})$ for an unlabelled dynamics $\mathcal{G}$ is a fairly new research direction, introduced in [18]. It is however very natural: as described in Subobjective 2.a, the interaction graph is a key parameter and many dynamical properties are invariant by isomorphism (number of fixed and periodic points, length of limit cycles and transient phases...). This setting allows many classical results to be expressed (e.g. if $\mathcal{G}$ corresponds to an additive dynamics, then $G(\mathcal{G})$ contains an interaction graph which is a disjoint union of companion digraphs [78]) but at the same time it brings many new perspectives, for instance: which are the dynamics $\mathcal{G}$ such that the set $G(\mathcal{G})$ is small (the dynamics needs specific interactions), large (the dynamics is rather ubiquitous), contains degree-bounded interaction graph (the dynamics can be produced with simple local computations)? We have started providing partial answers [18, 19], and pushing this direction further could lead to a deep renewal of the study of relationships between interaction graphs and dynamical properties. Tools come from various domains. For instance, we prove in [18] that $G(\mathcal{G})$ always contains an interaction graph with small in-degree using a difficult theorem from additive number theory [79].

Subobjective 2.d – Update modes. Update modes have been a central subject for the applicative aspects of AN theory [2, 3] (see [21] for an overview), and they are known to have a gradation of effects on the limit dynamics [67]. However, by another fundamental observation by Robert, fixed points are invariant under any block-sequential update schedule (where each automaton is updated exactly once during each step). Limit cycles, however, could differ, in a way characterized in [80, 81] for very simple architectures (cycles, with an even or odd number of negative arcs). The computational complexity approach is then employed to **explain why the pursue of such precise characterization is locked**: the corresponding decision problems are NP and Σ_2^{P}-complete [82, 5]. Counting the number of possible update modes of a given family is sometimes a computationally expensive task (#P-complete for block-sequential), and is sometimes given by nice formulas and enumerations (for block-parallel) [83, 22]. Such results give **a frame to further explorations**.

Through computational complexity issues including update modes, we have offered some **intuitive understanding** (even though the constructions are technical): on quantifier alternation in the question (to jump higher in PH [5]), on the possibility to embed arbitrary computation in substeps (PSPACE-completness [84]), and on the conversion among degrees of liberty in the model (local rules versus update modes [23]).

Objective 3 – Go beyond automata networks.

Subobjective 3.a – Simulation. The concept of reduction is a pillar of computer science used to classify problems (both in computability and complexity theory). When considering natural models of computation like CAs, several notions of *intrinsic simulations* have been studied [85] to classify them as dynamical systems. These notions are strong enough to preserve dynamical properties and automatically deduce reductions between several associated decision problems, but also general enough to obtain many simulation results and give rise to *intrinsic universality*. Recently, this approach was adapted to the model of ANs [86] with a focus on computational complexity. The above notions are useful to get lower bounds but are **strictly limited to a single model**. Simulation results can also be established between variations on a model: for instance, when changing the update schedule of ANs [87, 88]. However, to our knowledge, the corresponding notions of simulation were not studied in relation to computational complexity of decision problems. A **unifying framework of simulation** for finite dynamical systems is still missing.

Subobjective 3.b – Algebra of finite dynamical systems. **The literature on this domain is still scattered**, although direct product is a standard graph operator (Kronecker product in terms of adjacency matrices) [89].

The domain of succinct graph representation is typically interested in the transfer of complexity bounds from concrete (or explicit) to succinct graphs. We do not follow this trajectory, except for potential **converse transfers** from Rice-like PSPACE-hardness to (N)L-hardness. Various algorithms solving polynomial equations have been developed, whose worst case complexities are exponential, but show good performance in practice [35]. The **unroll of a dynamics** (family of trees constructed from limit cycles backward) has shown to be a key notion to study compositions, because tree-like dynamics (connected and with a unique fixed point) behave nicely with respect to + and · [90].

Building up on FANs project (ANR-18-CE40-0002). 5 members of our team were already members of the ANR JCJC project *Foundations of Automata Networks*, coordinated by Sylvain Sené from 2018 to 2022. This project was very fruitful (see the website for a detailed publication list), settling some important problems of the theory of automata networks, among which the complexity of some counting problems of fixed points and limit cycles, or the demonstration of some computational properties linked to the sensitivity to synchronism. The convenience and intuitiveness of our general proofs emphasized the relevance of notions of simulation, which we propose to understand further in our proposal.

On top of that, from the FANs project emerged very broad complexity results that were stated in the fashion of Rice's theorem. Unlike the previously mentioned results, even **the question itself was unexpected** in its form. One consequence is that the 5-persons team did not master all the topics that this family of questions relates to (finite model theory, tree automata, graph theory, combinatorics, circuit complexity, arithmetics…). Hence the idea of pushing it further, with an expanded consortium.

3 References

[1] H.G. Rice. "Classes of recursively enumerable sets and their decision problems". *Trans. AMS* (1953).
[2] S.A. Kauffman. "Metabolic stability and epigenesis in randomly constructed genetic nets". *Theoretical Biology* 22 (1969).
[3] R. Thomas. "Boolean formalization of genetic control circuits". *Theoretical Biology* 42.3 (1973).
[4] N. Alon. "Asynchronous threshold networks". *Graphs and Combinatorics* 1 (1985).
[5] F. Bridoux, C. Gaze-Maillot, K. Perrot, and S. Sené. "Complexity of Limit-Cycle Problems in Boolean Networks". *SOFSEM*. 2021.
[6] K. Perrot. "Études de la complexité algorithmique des réseaux d'automates". HDR. Aix Mars. Uni., 2022.
[7] G. Gamard, P. Guillon, K. Perrot, and G. Theyssier. "Rice-Like Theorems for Automata Networks". *STACS*. LIPIcs 187. 2021.
[8] H. Galperin. "Succinct Representations of Graphs". PhD. Princeton University, 1982.
[9] G. Gamard, P. Guillon, K. Perrot, and G. Theyssier. "Hardness of monadic second-order formulae over succinct graphs". *Preprint* (2023).
[10] J.L. Balcázar, A. Lozano, and J. Torán. "The complexity of algorithmic problems on succinct instances". *Computer Science* (1992).
[11] F. Bridoux, A. Durbec, K. Perrot, and A. Richard. "Complexity of fixed point counting problems in Boolean Networks". *J. of Computer and System Sciences* 126 (2022).
[12] T. Eiter, K. Makino, and G. Gottlob. "Computational aspects of monotone dualization: A brief survey". *Discrete Applied Mathematics* 156 (2008).
[13] N. Le Novère. "Quantitative and logic modelling of molecular and gene networks". *Nature Reviews Genetics* 16 (2015).
[14] N. Robertson, P. Seymour, and R. Thomas. "Permanents, Pfaffian orientations, and even directed circuits". *Annals of Mathematics* 150.3 (1999).
[15] J. Aracena. "Maximum Number of Fixed Points in Regulatory Boolean Networks". *Bulletin of Mathematical Biology* 70.5 (2008).
[16] M. Gadouleau, A. Richard, and S. Riis. "Fixed points of Boolean networks, guessing graphs, and coding theory". *SIAM J. on Discrete Maths* 29.4 (2015).
[17] E. Goles, P. Montealegre, M. Rios-Wilson, and G. Theyssier. "On the Impact of Treewidth in the Computational Complexity of Freezing Dynamics". *CiE*. 2021.
[18] F. Bridoux, K. Perrot, A. Picard Marchetto, and A. Richard. "Interaction graphs of isomorphic automata networks I". *J. of Computer and System Sciences* 138 (2023).
[19] J. Aracena, F. Bridoux, P. Guillon, K. Perrot, A. Richard, and G. Theyssier. "On the Dynamics of Bounded-Degree Automata Networks". *AUTOMATA explor.* 2023.
[20] F. Robert. "Itérations sur des ensembles finis et AC contractants". *Linear Algebra and its App.* 29 (1980).

[21] L. Paulevé and S. Sené. “Boolean networks and their dynamics: the impact of updates”. *Systems biology modelling and analysis* (2022).
[22] K. Perrot, S. Sené, and L. Tapin. “Complexity of Boolean automata networks under block-parallel update modes”. *SOFSEM*. 2024.
[23] M. Rios-Wilson and G. Theyssier. “Intrinsic Universality in Automata Networks III: On Symmetry versus Asynchrony”. *Preprint* (2022).
[24] K. Sutner. “On the Computational Complexity of Finite Cellular Automata”. *J. of Computer and System Sciences* 50.1 (1995).
[25] A. Dennunzio, E. Formenti, L. Manzoni, and A. E. Porreca. “Complexity of the dynamics of reaction systems”. *Information and Computation* 267 (2019).
[26] L. Adleman, Q. Cheng, A. Goel, M.D. Huang, D. Kempe, P.M. de Espanés, and P.W.K. Rothemund. “Combinatorial optimization problems in self-assembly”. *STOC*. 2002.
[27] M. Ota and S. Seki. “Rule set design problems for oritatami systems”. *Theoretical Computer Science* 671 (2017).
[28] M.J.P. Jiménez, A.R. Jiménez, and F.S. Caparrini. “Complexity classes in models of cellular computing with membranes”. *Natural Computing* 2 (2003).
[29] S. Perifel. *Complexité algorithmique*. Ellipses, 2014.
[30] C. Su, J. Pang, and S. Paul. “Towards Optimal Decomposition of Boolean Networks”. *IEEE/ACM Trans. Comput. Biol. Bioinformatics* 18 (2019).
[31] K. Perrot, P. Perrotin, and S. Sené. “On Boolean automata networks (de)composition”. *Fundamenta Informaticae* 181 (2021).
[32] K. Perrot, P. Perrotin, and S. Sené. “Optimising attractor computation in Boolean automata networks”. *LATA*. 2021.
[33] A. Dennunzio, B. Dorigatti, E. Formenti, L. Manzoni, and A. E. Porreca. “Polynomial Equations over Finite, Discrete-Time Dynamical Systems”. *ACRI*. 2018.
[34] F. Doré, E. Formenti, and A. E. Porreca S. Riva. “Decomposition and factorisation of transients in Functional Graphs”. *Preprint* (2022).
[35] S. Riva. “Factorisation of discrete dynamical systems”. PhD. Univ. Côte d’Azur, 2022.
[36] L. Fortnow and R. Santhanam. “Robust simulations and significant separations”. *Information and Computation* 256 (2017).
[37] P. Floréen and P. Orponen. “On the computational complexity of analyzing Hopfield nets”. *Complex Systems* 3 (1989).
[38] P. Orponen. “Neural networks and complexity theory”. *MFCS*. 1992.
[39] S. Kosub. “Dichotomy Results for Fixed-Point Existence Problems for Boolean Dynamical Systems”. *Mathematics in Computer Science* 1 (2008).
[40] J. Chalopin and V. Chepoi. “1-Safe Petri nets and special cube complexes: equivalence and applications”. *ACM Tr. on Computational Logic* 20 (2019).
[41] L. Bozyk, O. Defrain, K. Okrasa, and M. Pilipczuk. “On objects dual to tree-cut decompositions”. *Journal of Combinatorial Theory, Series B* 157 (2022).
[42] N. Fijalkow and B. Monmege. “Games with Payoffs”. *Games on Graphs*. Ed. by N. Fijalkow. 2023.
[43] P. Ohlmann, M. Pilipczuk, W. Przybyszewski, and S. Toruńczyk. “Canonical Decompositions in Monadically Stable and Bounded Shrubdepth Graph Classes”. *ICALP*. 2023.
[44] J. Gajarský, N. Mählmann, R. McCarty, P. Ohlmann, M. Pilipczuk, W. Przybyszewski, S. Siebertz, M. Sokołowski, and S. Toruńczyk. “Flipper Games for Monadically Stable Graph Classes”. *ICALP*. 2023.
[45] J. Chalopin, M. Changat, V. Chepoi, and J. Jacob. “First-order logic axiomatization of metric graph theory”. *Theoretical Computer Science* 993 (2024).
[46] H. Galperin and A. Wigderson. “Succinct representations of graphs”. *Information and Control* 56 (1983).
[47] C.H. Papadimitriou and M. Yannakakis. “A note on succinct representations of graphs”. *Information and Control* 71 (1986).
[48] J.L. Balcázar. “The complexity of searching succinctly represented graphs”. *ICALP*. 1995.
[49] B. Courcelle. “The monadic second-order logic of graphs. I. Recognizable sets of finite graphs”. *Information and Computation* 85 (1990).
[50] W. Thomas. “Ehrenfeucht Games, the Composition Method, and the Monadic Theory of Ordinal Words”. *Structures in Logic and Computer Science*. 1997.
[51] A. Blumensath, T. Colcombet, and C. Löding. “Logical theories and compatible operations”. *Logic and Automata: History and Perspectives*. Vol. 2. 2008.
[52] A.L. Semenov. “Decidability of monadic theories”. *MFCS*. 1984.
[53] D. Berwanger and A. Blumensath. “The Monadic Theory of Tree-like Structures”. *Automata Logics, and Infinite Games*. 2002.
[54] J. Kari. “Rice’s theorem for the limit sets of cellular automata”. *Theoretical Computer Science* 127 (1994).
[55] P. Guillon and G. Richard. “Revisiting the Rice Theorem of Cellular Automata”. *STACS*. 2010.

[56] M. Delacourt. “Rice’s Theorem for mu-Limit Sets of Cellular Automata”. *ICALP*. 2011.
[57] B. Borchert and F. Stephan. “Looking for an Analogue of Rice’s Theorem in Circuit Complexity Theory”. *Mathematical Logic Quarterly* 46 (2000).
[58] L.A. Hemaspaandra and J. Rothe. “A second step towards complexity-theoretic analogs of Rice’s Theorem”. *Theoretical Computer Science* 244 (2000).
[59] L.A. Hemaspaandra and M. Thakur. “Lower bounds and the hardness of counting properties”. *Theoretical Computer Science* 326 (2004).
[60] L. Mendoza, D. Thieffry, and E. R. Alvarez-Buylla. “Genetic control of flower morphogenesis in Arabidopsis thaliana: a logical analysis”. *Bioinformatics* 15 (1999).
[61] Y. Xiao. “A Tutorial on Analysis and Simulation of Boolean Gene Regulatory Network Models”. *Curr Genomics* 10 (2009).
[62] G.J. Hickmann and T.C. Hodgman. “Inference of gene regulatory networks using boolean-network inference methods”. *Bioinformatics and Computational Biology* 7 (2009).
[63] Z. Pusnik, M. Mraz, N. Zimic, and M. Moskon. “Review and assessment of Boolean approaches for inference of gene regulatory networks”. *Heliyon* 8 (2022).
[64] A. Poindron. “A general model of binary opinions updating”. *Math. Social Sciences* 109 (2021).
[65] R. Thomas. “On the relation between the logical structure of systems and their steady states or sustained oscillations”. *Numerical methods* 9 (1981).
[66] A. Richard. “Negative circuits and sustained oscillations in asynchronous automata networks”. *Advances in Applied Mathematics* 44 (2010).
[67] S. Sené. “Sur la bio-informatique des réseaux d’automates”. Univ. Evry Val d’Essonne, 2012.
[68] M. Gadouleau. “On the Rank and Periodic Rank of Finite Dynamical Systems”. *Elec. J. of Combinatorics* 25.3 (2018).
[69] A. Richard. “Positive and negative cycles in Boolean networks”. *Theoretical Biology* 463 (2019).
[70] M. Gadouleau. “On the influence of the interaction graph on a finite dynamical system”. *Natural Computing* 19 (2020).
[71] S. Riis. “Information Flows, Graphs and their Guessing Numbers”. *Elec. J. of Combinatorics* 14 (2007).
[72] J. Aracena, A. Richard, and L. Salinas. “Number of Fixed Points and Disjoint Cycles in Monotone Boolean Networks”. *SIAM J. on Discrete Mathematics* 31 (2017).
[73] A. Naldi et al. “The CoLoMoTo Interactive Notebook: Accessible and Reproducible Computational Analyses for Qualitative Biological Networks”. *Frontiers in Physiology* (2018). See the website.
[74] Y. Crama and P.L. Hammer. *Boolean Functions: Theory, Algorithms, and Applications*. Cambridge University Press, 2011.
[75] G. Bagan, A. Bonifati, and B. Groz. “A trichotomy for regular simple path queries on graphs”. *J. of Computer and System Sciences* 108 (2020).
[76] F. Madelaine and B. Martin. “A Tetrachotomy for Positive First-Order Logic without Equality”. *LICS*. 2011.
[77] O. Gerasimova, S. Kikot, A. Kurucz, V. Podolskii, and M. Zakharyaschev. “A tetrachotomy of ontology-mediated queries with a covering axiom”. *Artificial Intelligence* 309 (2022).
[78] B. Elspas. “The theory of autonomous linear sequential networks”. *IRE Tr. on Circuit Theory* 6 (1959).
[79] N. Alon and M. Dubiner. “A lattice point problem and additive number theory”. *Combinatorica* 15 (1995).
[80] E. Goles and M. Noual. “Block-sequential update schedules and Boolean automata circuits”. *AUTOMATA*. 2010.
[81] J. Demongeot, M. Noual, and S. Sené. “Combinatorics of Boolean automata circuits dynamics”. *Discrete Applied Mathematics* 160 (2012).
[82] J. Aracena, L. Gómez, and L. Salinas. “Limit cycles and update digraphs in Boolean networks”. *Discrete Applied Mathematics* 161 (2013).
[83] C. Noûs, K. Perrot, S. Sené, and L. Venturini. “#P-completeness of counting update digraphs, cacti, and series-parallel decomposition method”. *CiE*. 2020.
[84] K. Perrot, S. Sené, and L. Tapin. “Complexity of Boolean automata networks under block-parallel update modes”. *SAND (to appear)*. 2024.
[85] M. Delorme, J. Mazoyer, N. Ollinger, and G. Theyssier. “Bulking II: Classifications of cellular automata”. *Theoretical Computer Science* 412 (2011).
[86] M. Rios-Wilson and G. Theyssier. “Intrinsic Universality in Automata Networks I: Families and Simulations”. *Theoretical Computer Science* (2024).
[87] T. Melliti, D. Regnault, A. Richard, and S. Sené. “Asynchronous simulation of Boolean networks by monotone Boolean networks”. *ACRI*. 2016.
[88] F. Bridoux, P. Guillon, K. Perrot, S. Sené, and G. Theyssier. “On the cost of simulating a parallel Boolean automata network with a block-sequential one”. *TAMC*. 2017.
[89] R. Hammack, W. Imrich, and S. Klavzar. *Handbook of Product Graphs*. CRC Press, Inc., 2011.
[90] É. Naquin and M. Gadouleau. “Factorisation in the semiring of finite dynamical systems”. *Preprint* (2022).