Fuzzy Sets and Systems 530 (2026) 109749

Contents lists available at ScienceDirect

Fuzzy Sets and Systems

journal homepage: www.elsevier.com/locate/fss

A Gödel logic descriptor for chains of regular languages

Stefano Aguzzoli [a], Brunella Gerla [b], Sarah Nastasi [b,*]

[a] DI, University of Milan, Via Celoria 18, Milano, 20133, Italy
[b] DiSTA, University of Insubria, Pad. Rossi, Via O. Rossi 9, Varese, 21100, Italy

ARTICLE INFO

Keywords:
Gödel propositional logic
Regular languages
Categorical duality
Finite forests
Automata

ABSTRACT

Fortresses constitute a class of language descriptors completely based only on two fundamental concepts of propositional logic: the notion of logical consequence and the notion of substitution. Fortresses based on classical propositional logic precisely recognise the class of regular languages.

In this paper, we characterise the formal languages obtained by replacing, in fortresses, the notion of logical consequence in classical logic with the one in Gödel infinitely-valued logic. We prove that fortresses in Gödel logic exactly recognise finite chains of inclusions of regular languages. To prove this, we make use of a Stone's type categorical duality between the algebraic semantics of Gödel propositional logic and the category of finite forests and open maps.

1. Introduction

Gödel (or Gödel-Dummett) propositional logic [1,2] is the extension of intuitionistic propositional logic with the axiom of *prelinearity* $(\varphi \to \psi) \vee (\psi \to \varphi)$, or, equivalently, it is one of the major mathematical fuzzy logics, as it is obtained by extending either Esteva and Godo's Monoidal t-norm based logic MTL, or Hájek's Basic fuzzy logic BL with the axiom of *idempotence* of the monoidal conjunction $\varphi \to (\varphi \& \varphi)$ [3,4]. As a matter of fact, Gödel propositional logic is the logic of the minimum t-norm $\min\{x, y\}$ and its residuum, and as such it is one of the three fundamental logics based on continuous t-norms. One of the main tasks faced by mathematical fuzzy logic is to understand the semantic features, similarities and differences among the infinitely many propositional logics, which generalise classical Boolean logic over the two-element truth-value set $\{0, 1\}$ to the much richer setting of the real unit interval $[0, 1]$ as the set of truth-values. In this perspective, a useful approach to a many-valued logic L consists in generalising to L the classical concepts of Boolean logic using some reasonably robust conceptual framework, beyond the mere enlargement of the truth-value set.

We recall here two milestones in the study of classical propositional logic: its algebraic semantics, given by the variety of Boolean algebras, and the categorical dual equivalence between Boolean algebras and the category of Stone spaces, that is, totally disconnected compact Hausdorff topological spaces [5]. Notice that Stone's duality, when restricted to the subcategory of *finite* Boolean algebras, states that the latter is dually equivalent to the category of finite sets and maps between them.

Gödel propositional logic too, is *algebraisable* in the sense of Lindenbaum-Tarski and Blok-Pigozzi [6], and its algebraic semantics is given by the variety of Gödel algebras (which contains Boolean algebras as a subvariety). Drawing a parallel with the Boolean case, the category of finite Gödel algebras is dually equivalent to the computationally manageable category of finite forests and order-preserving open maps between them (which clearly contains the category of finite sets as a full subcategory).

The duality between finite Gödel algebras and finite forests has been used in several directions to generalise concepts from classical logic to Gödel logic. For instance, duality has been used to describe coproducts among finite Gödel algebras [7,8], to characterise homomorphisms and automorphisms of finite Gödel algebras [9,10], to define probabilities over events described by formulas of

* Corresponding author.
E-mail addresses: stefano.aguzzoli@unimi.it (S. Aguzzoli), brunella.gerla@uninsubria.it (B. Gerla), snastasi@uninsubria.it (S. Nastasi).

https://doi.org/10.1016/j.fss.2025.109749
Received 21 July 2025; Received in revised form 28 November 2025; Accepted 22 December 2025
Available online 29 December 2025
0165-0114/

Gödel logic [11,12]. As far as we know, this approach has not yet been pursued for classical models of computation, such as the automata for recognising formal languages. In this paper we shall generalise the approach introduced in [13] to develop a notion of automaton based on Gödel logic.

The authors of [13] propose a descriptor of formal languages which is based *only* on two fundamental concepts of classical propositional logic: the notion of logical consequence (of a formula φ from a theory Θ), and the notion of substitution of a variable with a formula (a map σ from the set of variables to the set of formulas, which is canonically extended to a map from formulas to formulas). As a matter of fact, they define the notion of *classical fortress* as a triple formed by a theory, a formula, and a bunch of substitutions (one for each letter in an alphabet Σ), and show that a very natural relation between these logical ingredients and an input string $s \in \Sigma^*$ allows to characterise each regular language as the set of words satisfying this relation for a specific fortress, and, by converse, for each fortress, the set of words satisfying the relation constitutes a regular language.

To prove the correspondence *fortress-regular language*, the paper [13] actually uses algebraic semantics and duality to prove a correspondence between fortresses and deterministic finite state automata, as is well known that these computational devices exactly recognise the class of regular languages.

In this paper we generalise the approach formulated in [13] from classical propositional logic to Gödel logic, using the algebraic semantics of finite Gödel algebras, and the dually equivalent category of finite forests, to associate bijectively with each fortress a computational finite-state device that we call Gödel automaton. We prove that Gödel automata recognise exactly the finite sequences of inclusions of regular languages.

We stress that our notion of Gödel fortress only differs from the notion of classical fortress as it replaces classical logical consequence with logical consequence in Gödel logic: that is to say, any classical fortress is a Gödel fortress when interpreted with the Gödel logic consequence relation. This observation indicates that the structure of a Gödel automaton arises *naturally* from Gödel logic and its logical consequence, and therefore, also the class of formal languages recognised by such devices (the finite inclusion chains of regular languages) stands to the class of regular languages as Gödel logic stands to classical Boolean logic. Furthermore, the approach we have used to obtain the above characterisation can be generalised in principle to every propositional logic in the same language of classical logic.

The main motivations of this paper lie in characterising the formal languages recognised by the purely logic oriented definition of Gödel fortresses, and we introduce Gödel automata as instruments to obtain this characterisation. This notwithstanding, Gödel automata are of interest in themselves, as they constitute another class of automata that recognise fuzzy languages as defined, for instance, in [14].

The paper is organised as follows: In Section2, first we briefly recall the basics of regular languages and automata, and a description of the main concepts in Gödel algebra and duality follows. In Section 3 we summarise the main concepts and results of the paper [13] for the Boolean case. Our main definitions and results, that is Gödel automata, Gödel fortresses and their relation with chains of regular languages, are discussed in Sections 4 and 5. In Section 6 we discuss some features of fortresses, their relations with the Büchi-Elgot-Trakhtenbrot theorem and the relations between Gödel automata and fuzzy automata.

2. Preliminaries

2.1. Languages and automata

We collect in this section some definitions and results on regular languages and finite deterministic automata that we will use in the paper. We refer the reader to [15] for further details.

An *alphabet* Σ is a finite set of elements that we identify as *letters* or *symbols*. Each ordered concatenation of $n \in \mathbb{N}$ letters is a *string* (or *word*) of length n. We take ε as a special character representing the string of zero length, also called the *empty string*. The concatenation of letters naturally extends to strings of any length, such that for any two strings s and t, the concatenation $s \circ t = st$ results in a string of length $|st| = |s| + |t|$. A set of possibly infinitely many strings is a *(formal) language*. Similarly to what we do for letters and strings, the concatenation of two languages $\mathscr{L}_1$ and $\mathscr{L}_2$ is the set $\mathscr{L}_1 \circ \mathscr{L}_2 = \mathscr{L}_1\mathscr{L}_2 = \{xy \mid x \in \mathscr{L}_1, y \in \mathscr{L}_2\}$.
Besides concatenation, there are other relevant operations for strings and languages like the power and the Kleene star, which are built on it. Given any string s in any language $\mathscr{L}$ and $n \in \mathbb{N}$, we define the *power* as $s^n = \varepsilon$ if $n = 0$ while $s^n = s \circ s^{n-1}$, otherwise. Analogously, $\mathscr{L}^0 = \{\varepsilon\}$, while $\mathscr{L}^n = \mathscr{L} \circ \mathscr{L}^{n-1}$, otherwise. The *Kleene star* (Kleene closure, or Kleene operator) is defined by $\mathscr{L}^* = \bigcup_{n=0}^{\infty} \mathscr{L}^n$. In particular, if $\mathscr{L} = \{s\}$ we use the notation s^* instead of $\{s\}^*$. The aforementioned operations let us specify the family of regular languages. A language $\mathscr{L}$ over an alphabet Σ is a *regular language* if and only if $\mathscr{L} = \emptyset \mid \{\epsilon\} \mid \{x\} \mid \mathscr{L} \cup \mathscr{L} \mid \mathscr{L} \circ \mathscr{L} \mid \mathscr{L}^*$, with $x \in \Sigma$.

A *deterministic finite automaton* (DFA) on an alphabet $\Sigma \neq \emptyset$ is a quadruple $\mathscr{A} = (Q, \delta, i, F)$, where:

1. Q is a finite set of *states*
2. $\delta : Q \times \Sigma \to Q$ is the *transition function*
3. $i \in Q$ is the *initial state*
4. $F \subseteq Q$ is the set of *final states*.

Notice that in our definition, the automaton is *complete*, that is, δ is a *total* function.

Graphically, each DFA is represented as a directed network where states are drawn as circles. They are marked with a small entering arrow if initial, and they are circled twice if final. Each transition $\delta(q, a) = p$ is depicted as $q \xrightarrow{a} p$, where a directed arrow from the state q to p (possibly coinciding) is labelled with the input letter $a \in \Sigma$. The transition relation δ naturally extends to $\delta^* : Q \times \Sigma^* \to Q$

by setting (i) $\delta^*(q,\varepsilon) = q$ and (ii) $\delta^*(q,as) = \delta^*(\delta(q,a),s)$ for every $a \in \Sigma$, $s \in \Sigma^*$ and $q \in Q$. A state q is said to be *reachable* if there exists a string $s \in \Sigma^*$ such that $\delta^*(i,s) = q$. We say that an automaton is *reachable* when every state is reachable.

Definition 1. A string $s \in \Sigma^*$ is *accepted* by the DFA $\mathscr{A} = (Q,\delta,i,F)$ if and only if $\delta^*(i,s) \in F$. A language $\mathscr{L}$ is accepted by $\mathscr{A}$ if and only if $\mathscr{A}$ accepts exactly all the strings in $\mathscr{L}$. We denote by $\mathscr{A}(\mathscr{L})$ an automaton accepting a language $\mathscr{L}$, while the language accepted by an automaton $\mathscr{A}$ is denoted by $\mathscr{L}(\mathscr{A})$.

We recall that every regular language can be recognised by a reachable deterministic finite automaton, and any language recognised by a DFA is regular.

Finally, we introduce an important operation between automata, which will be used in the proof of Theorem 4.

Definition 2. The *intersection automaton* of two deterministic automata $\mathscr{A}_1 = (Q_1,\delta_1,i_1,F_1)$ and $\mathscr{A}_2 = (Q_2,\delta_2,i_2,F_2)$ on an alphabet Σ, denoted by $\mathscr{A}_1 \cap \mathscr{A}_2$, is the reachable version of the automaton $\mathscr{A} = (Q_\cap,\delta_\cap,i_\cap,F_\cap)$ such that:

1. $Q_\cap = Q_1 \times Q_2$
2. $\delta_\cap((q_1,q_2),a) = (\delta_1(q_1,a),\delta_2(q_2,a))$, with $a \in \Sigma$, $q_1 \in Q_1$, $q_2 \in Q_2$
3. $i_\cap = (i_1,i_2)$
4. $F_\cap = F_1 \times F_2$.

The intersection automaton is hence obtained from the two original automata, $\mathscr{A}_1$ and $\mathscr{A}_2$, by making them flow together in parallel.

Proposition 1. *Let $\mathscr{L}(\mathscr{A}_1)$ and $\mathscr{L}(\mathscr{A}_2)$ be two regular languages accepted by automata $\mathscr{A}_1$ and $\mathscr{A}_2$, then $\mathscr{L}(\mathscr{A}_1 \cap \mathscr{A}_2) = \mathscr{L}(\mathscr{A}_1) \cap \mathscr{L}(\mathscr{A}_2)$.*

2.2. Gödel algebras

Gödel propositional logic is the axiomatic extension of intuitionistic propositional logic by the prelinearity axiom $(\varphi \to \psi) \vee (\psi \to \varphi)$ [2]. It also arises in the framework of fuzzy logics as the propositional logic associated with the minimum t-norm, obtained by extending Hájek's Basic Fuzzy Logic BL with the *idempotence* axiom $\varphi \to (\varphi \& \varphi)$ [3].

The algebraic counterparts of Gödel logic are Gödel algebras, see for instance [3,16]. A Gödel algebra $\mathbf{A} = (A,\vee,\wedge,\to,0,1)$ is an algebraic structure such that:

1. $(A,\vee,\wedge,0,1)$ is a bounded distributive lattice with 0 as its largest element and 1 as its smallest element;
2. for any $a,b,c \in A$: $a \wedge b \le c$ iff $a \le b \to c$;
3. (Prelinearity equation) for any $a,b \in A$: $(a \to b) \vee (b \to a) = 1$.

In other words, Gödel algebras are prelinear Heyting algebras. We also recall to the reader that in every Gödel algebra the order $\le$ of the lattice may be recovered from the implication, i.e. $a \le b$ iff $a \to b = 1$.

In what follows, the signature $(A,\wedge,\to,0)$ will be employed for Gödel algebras without loss of generality, since the remaining operations can be defined from the given ones, being $a \vee b = ((a \to b) \to b) \wedge ((b \to a) \to a)$ and $1 = 0 \to 0$. Further, as it is customary, we define $\neg a$ as $a \to 0$, and $a \leftrightarrow b$ as $(a \to b) \wedge (b \to a)$.

A *filter* U of a Gödel algebra $\mathbf{A} = (A,\wedge,\to,0)$ is an upward closed subset of A such that if $a,b \in U$ then $a \wedge b \in U$. A filter U of $\mathbf{A}$ is *principal* if and only if it has the form $\{b \in A \mid a \le b\}$ for some element $a \in A$, called the *generator* of U. In this case we write $U = \langle a \rangle$. A filter $\mathfrak{p}$ of $\mathbf{A}$ is *prime* if and only if it is proper, that is, $\mathfrak{p} \subsetneq A$, and for all $a,b \in A$ either $a \to b \in \mathfrak{p}$ or $b \to a \in \mathfrak{p}$. A filter is *maximal* if it is not properly included in any other filter. Maximal filters are prime, but in general prime filters are not maximal. The set of all prime filters of $\mathbf{A}$, ordered by reverse inclusion, is called the *prime spectrum* $Spec(\mathbf{A})$ of $\mathbf{A}$. If $\mathbf{A}$ is a finite Gödel algebra, for every prime filter $\mathfrak{p} \in Spec(\mathbf{A})$ the (finite) set $\{\mathfrak{p}' \in Spec(\mathbf{A}) \mid \mathfrak{p} \subseteq \mathfrak{p}'\}$ is totally ordered by inclusion: this is one of the key observations leading to the duality between finite Gödel algebras and finite forests, as described in Section 2.3 (see also [2]).

A *join-irreducible* element in a Gödel algebra $\mathbf{A}$ is a join-irreducible element in the underlying lattice, that is an element $0 \neq j \in A$ such that if there are $a,b \in A$ with $j = a \vee b$ then either $a = j$ or $b = j$. If A is a finite Gödel algebra, for every $a \in A$ there are join-irreducible elements $j_1,\ldots,j_s$ such that $a = \bigvee_{i=1}^{s} j_i$. We observe that, in any finite Gödel algebra $\mathbf{A}$, the prime spectrum $Spec(\mathbf{A})$ is order-isomorphic with the poset $\mathrm{JI}\,\mathbf{A}$ of the join-irreducible elements, since each prime filter is principal and it is generated by a join-irreducible element (and distinct join-irreducible elements generate distinct filters). The map sending each prime filter $\mathfrak{p}$ of $\mathbf{A}$ to its minimum element $j_\mathfrak{p} = \min \mathfrak{p}$ is an order isomorphism between $(Spec(\mathbf{A}),\supseteq)$ and $(\mathrm{JI}\,\mathbf{A},\le)$. Its inverse map sends each join-irreducible element j to the prime filter $\langle j \rangle = \{a \in A \mid a \ge j\}$.

The main example of Gödel algebra is the standard Gödel algebra

$$([0,1],\min,\to_{\min},0),$$

where $a \to_{\min} b = 1$ if $a \le b$, while $a \to_{\min} b = b$ otherwise. Indeed, Gödel algebras form a locally finite variety $\mathbb{G}$ that is generated by the standard Gödel algebra. Further, since every Gödel algebra satisfies the prelinearity axiom, every Gödel algebra is a subdirect product of linearly ordered Gödel algebras [2].

The set $Form(n)$ of formulas of Gödel logic with n variables is inductively built from the constant 0 and the set $Var(n) = \{x_1,\ldots,x_n\}$ using binary connectives $\wedge$ and $\to$.

An evaluation is a map $v : Var(n) \to [0,1]$ canonically extended to a map $v : Form(n) \to [0,1]$ by defining $v(0) = 0$, $v(\varphi \wedge \psi) = \min(v(\varphi),v(\psi))$ and $v(\varphi \to \psi) = 1$ if $v(\varphi) \le v(\psi)$ while $v(\varphi \to \psi) = v(\psi)$ otherwise. If $v(\varphi) = 1$ we write $v \vDash \varphi$ and we say that v satisfies the formula φ. Given a set of formulas Θ, we have $\Theta \vDash \varphi$ if every evaluation that satisfies all formulas of Θ also satisfies φ. We write $\psi \vDash \varphi$ to mean $\{\psi\} \vDash \varphi$.

A *theory* Θ over $Var(n)$ is a set of formulas in $Form(n)$ such that if $\Theta \vDash \varphi$ then $\varphi \in \Theta$.[1] A theory is *prime* if $\varphi \vee \psi \in \Theta$ implies that either $\varphi \in \Theta$ or $\psi \in \Theta$. A theory Θ is *principal* if there exists a formula φ such that $\Theta = \{\psi \mid \varphi \vDash \psi\}$. Let us notice that every finite theory of Gödel logic is principal. A theory is *maximal* if it is not properly included in any other consistent theory.

Two formulas φ and ψ are logically equivalent, and we write $\varphi \equiv \psi$, if $v(\varphi) = v(\psi)$ for every evaluation v. Logical equivalence is a congruence with respect to the operations induced by connectives (that is $[\varphi] * [\psi] = [\varphi * \psi]$ for $* \in \{\wedge, \rightarrow\}$). The algebra

$$\mathcal{G}_n = (Form(n)/\equiv, \wedge, \rightarrow, [0])$$

is the *free Gödel algebra* over n free generators $\mathcal{G}_n$. Note that $\mathcal{G}_n$ is a finite Gödel algebra.

As join-irreducible elements of the Gödel algebra $\mathcal{G}_n$ are equivalence classes of formulas, we recall here a normal form theorem (see for instance [16–18]) that allows us to pick a formula (that we shall call a *minterm*) in each join-irreducible element of $\mathcal{G}_n$.

Let $[0,1]^{[0,1]^n}$ be the Gödel algebra of all functions $f : [0,1]^n \to [0,1]$ equipped with pointwise defined operations (that is $(f * g)(t_1,\dots,t_n) = f(t_1,\dots,t_n) * g(t_1,\dots,t_n)$ for $* \in \{\wedge, \rightarrow\}$, and the bottom element is the function constantly 0). Since the standard Gödel algebra generates the whole variety $\mathbb{G}$, a standard universal algebra argument (see [19]) proves that $\mathcal{G}_n$ is isomorphic with the subalgebra of $[0,1]^{[0,1]^n}$ generated by the projections functions $x_i : (t_1,\dots,t_n) \mapsto t_i$. Following the characterisation of $\mathcal{G}_n$ as an algebra of $[0,1]$-valued functions given in [18], we introduce the notion of Gödel n-variate region as the set of points $(x_1,\dots,x_n) \in [0,1]^n$ satisfying an expression of the form

$$0 \prec_0 x_{\iota(1)} \prec_1 \cdots \prec_{n-1} x_{\iota(n)} \prec_n 1,$$

where $\prec_i \in \{<,=\}$ ($\prec_i$ must be $<$ for at least one index i) and $\iota : \{1,\dots,n\} \to \{1,\dots,n\}$ is a permutation. Given Gödel n-variate regions P, Q, we stipulate $P \sqsubseteq Q$ iff there is an index j such that P is obtained replacing in Q all $\prec_h$ with $=$ for all indices $h \geq j$. For instance, $(0 < x = y = 1) \sqsubset (0 < x < y = 1) \sqsubset (0 < x < y < 1)$. Clearly, $\sqsubseteq$ is a partial order relation. Let us write $GR(n)$ for the poset of all Gödel n-variate regions partially ordered by $\sqsubseteq$.

Proofs of the following statements can be found in [16–18].

Lemma 1. *For each $n \geq 0$, there is a poset isomorphism $\eta_n : GR(n) \to \mathrm{JI}\,\mathcal{G}_n$. Given the Gödel n-variate region $P \subseteq [0,1]^n$, satisfying $0 \prec_0 x_{\iota(1)} \prec_1 \cdots \prec_{n-1} x_{\iota(n)} \prec_n 1$, write $x_{\iota(0)}$ for 0 and $x_{\iota(n+1)}$ for 1. Then, the* minterm *m_P is the formula*

$$m_P = \bigwedge_{i=0}^{n} \mu_i,$$

where $\mu_i = x_{\iota(i)} \leftrightarrow x_{\iota(i+1)}$ iff $\prec_i$ is $=$, while $\mu_i = (x_{\iota(i+1)} \to x_{\iota(i)}) \to x_{\iota(i+1)}$ iff $\prec_i$ is $<$. It holds that $m_P \in \eta_n(P)$.

Remark 1. Some simplifications can be applied in writing minterms: for instance $(x \to 0) \to x$ is equivalent with $\neg\neg x$, and $0 \leftrightarrow x$ is equivalent with $\neg x$; further $(1 \to x) \to 1$ is equivalent with 1 and $x \leftrightarrow 1$ is equivalent with x. By a slight abuse of terminology we shall call *minterm* any formula belonging to a join-irreducible element of $\mathcal{G}_n$.

Example 1. There are eleven Gödel 2-variate regions in $GR(2)$:

$P_1 = \{(x,y) \mid 0 < x = y = 1\}$
$P_2 = \{(x,y) \mid 0 < y < x = 1\}$
$P_3 = \{(x,y) \mid 0 < y < x < 1\}$
$P_4 = \{(x,y) \mid 0 < x = y < 1\}$
$P_5 = \{(x,y) \mid 0 < x < y = 1\}$
$P_6 = \{(x,y) \mid 0 < x < y < 1\}$
$P_7 = \{(x,y) \mid 0 = x < y = 1\}$
$P_8 = \{(x,y) \mid 0 = x < y < 1\}$
$P_9 = \{(x,y) \mid 0 = x = y < 1\}$
$P_{10} = \{(x,y) \mid 0 = y < x = 1\}$
$P_{11} = \{(x,y) \mid 0 = y < x < 1\}$

with the following order relation:

$P_1 \sqsubseteq P_2 \sqsubseteq P_3$; $P_1 \sqsubseteq P_5 \sqsubseteq P_6$; $P_1 \sqsubseteq P_4$;
$P_7 \sqsubseteq P_8$; $P_{10} \sqsubseteq P_{11}$.

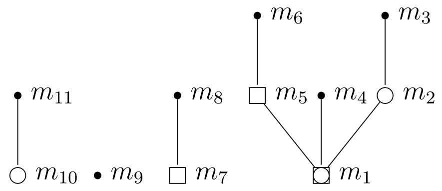
Let us determine the formulas of the minterms m_2, m_3 from the corresponding Gödel 2-variate regions P_2, P_3. According to Lemma 1, we should consider $m_2 = \mu_1 \wedge \mu_2 \wedge \mu_3$ such that $\mu_1 = (y \to 0) \to y$, $\mu_2 = (x \to y) \to x$ and $\mu_3 = x \leftrightarrow 1$. Then, since $\mu_1 \equiv \neg y \to y \equiv \neg\neg y$, $\mu_3 \equiv (x \to 1) \wedge (1 \to x) \equiv 1 \wedge x \equiv 1$ and $\mu_2 \wedge \mu_3 \equiv \mu_3$ (μ_2 equals x when $x < y$, 1 otherwise), we may also rewrite $m_2 = \neg\neg y \wedge x$. Similarly, $m_3 = \mu_1 \wedge \mu_2 \wedge \mu_4$ with $\mu_4 = (1 \to x) \to 1$. As above, the formula for the minterm can also be shortened as $m_3 = \neg\neg y \wedge ((x \to y) \to x)$, being $\mu_4 \equiv x \to 1 \equiv 1$. For a comprehensive list of minterms in $\mathcal{G}_2$, see Fig. 1 where the nodes of the poset $\mathrm{JI}\,\mathcal{G}_2$ are labelled by minterms as representatives for each join irreducible element.

Remark 2. $GR(n)$ (and then $\mathrm{JI}\,\mathcal{G}_n$ and $Spec(\mathcal{G}_n)$) can be effectively constructed by recursion on n as specified for instance in [8,16].

Lemma 2. *Each formula $\varphi \in Form(n)$ is logically equivalent to a disjunction of minterms: those minterms m such that $m \to \varphi$ is equivalent with 1 (or, equivalently, $[m] \leq [\varphi]$ in $\mathcal{G}_n$).*

In the following lemma, we collect some results about theories, minterms, and filters in the free Gödel algebra, see [16].

[1] In this paper we assume theories are deductively closed sets of formulas.



$$\begin{aligned} m_1 &= x \wedge y & m_7 &= y \wedge \neg x \\ m_2 &= x \wedge \neg\neg y & m_8 &= \neg\neg y \wedge \neg x \\ m_3 &= \neg\neg y \wedge ((x \to y) \to x) & m_9 &= \neg x \wedge \neg y \\ m_4 &= (x \to y) \wedge (y \to x) \wedge \neg\neg x & m_{10} &= x \wedge \neg y \\ m_5 &= y \wedge \neg\neg x & m_{11} &= \neg\neg x \wedge \neg y \\ m_6 &= \neg\neg x \wedge ((y \to x) \to y) \end{aligned}$$

Fig. 1. $\mathscr{F}_2$ labelled with minterms. Minterms m such that $m \vDash x$ are marked with white circles, while those for which $m \vDash y$ are marked with white squares.

Lemma 3. *Every prime theory Θ corresponds to a prime filter $\{[\varphi] \mid \varphi \in \Theta\}$ of $\mathscr{G}_n$, which is generated by the minterm m such that $\Theta = \{\varphi \mid m \vDash \varphi\}$ (we say that Θ is generated by m). Given a prime theory Θ generated by a minterm m there are prime theories $\Theta_1, \dots, \Theta_h$, respectively generated by minterms $m_1, \dots, m_h$ such that $\Theta = \Theta_h \subset \dots \subset \Theta_1$ and $[m] = [m_h] > \dots > [m_1]$. Θ_1 is a maximal theory, and the filter generated by $[m_1]$ is maximal.*

Example 2. Let us consider the minterms m_1, m_2, m_3 in Fig. 1, which are such that $m_1 \vDash m_2$, $m_2 \vDash m_3$ and $m_1 \vDash m_3$. If we consider their equivalence classes, we get the chain $[m_1] \le [m_2] \le [m_3]$ of join-irreducible elements of $\mathscr{G}_2$. Since $\Theta_i = \{\varphi \mid m_i \vDash \varphi\}$, then $\Theta_3 \subseteq \Theta_2 \subseteq \Theta_1$. Moreover, Θ_1 is maximal.

Lastly, we shall need the following notion. *Substitutions* are defined as functions $\sigma : Var(n) \to Form(n)$ and, for every formula $\varphi \in Form(n)$, we denote by $\varphi[\sigma]$ the formula obtained by replacing in φ all the occurrences of x_i with $\sigma(x_i)$. Composition of substitutions is denoted by $\circ$ and it acts so that $\varphi[\sigma \circ \tau] = (\varphi[\tau])[\sigma]$.

2.3. Duality for finite Gödel algebras

The dual equivalence between finite Gödel algebras and finite forests (and order-preserving open maps) is implicit in [2,20]. See [7,16] for details. It can also be obtained by specialising Esakia's duality [21] to finite prelinear Heyting algebras.

A *forest* is a poset $\mathscr{F}$ such that the downset $\downarrow x = \{y \in \mathscr{F} \mid y \le x\}$ of every element $x \in \mathscr{F}$ is totally ordered. The minimal elements of $\mathscr{F}$ are called *roots*. A *tree* is a forest with minimum.

An order-preserving map $f : F \to G$ between forests is *open* if for all $x, y \in F$ if $y \le f(x)$ then there is $z \le x$ such that $f(z) = y$. Equivalently, an open map carries downward-closed sets to downward-closed sets.

We shall write FF for the category whose objects are the *finite forests* and whose arrows are the order-preserving open maps between them.

It is possible to check that prelinearity implies that the prime spectrum of a Gödel algebra is a forest, that is the set $\{\mathfrak{p}' \in Spec(\mathbf{A}) \mid \mathfrak{p} \subseteq \mathfrak{p}'\}$ is always totally ordered by inclusion. Further, the intersection of two non-disjoint prime filters is again a prime filter. Equivalently, the meet of two join-irreducible elements is either [0] or again join-irreducible. This is a key observation for the following result. Let $Sub(\mathscr{F})$ be the set of all downward-closed subsets of $\mathscr{F}$.

Theorem 1. *[16, Thm. 4.2.1] The category of finite Gödel algebras and their homomorphisms $\mathbb{G}_{fin}$ is dually equivalent to the category of finite forests FF, via the following functors:*

$$\mathbf{Sub} : \mathsf{FF} \to \mathbb{G}_{fin},$$

and

$$\mathrm{Spec} : \mathbb{G}_{fin} \to \mathsf{FF},$$

where:

- *for each finite forest F, $\mathbf{Sub}\,F = (Sub(F), \cap, \to, \emptyset)$, where $H \to K = F \setminus \uparrow (H \setminus K)$ (for $\uparrow S$ being the upward closure of S), and for each order-preserving open map $f : F \to G$, the homomorphism $\mathbf{Sub}\,f : \mathbf{Sub}\,G \to \mathbf{Sub}\,F$ is defined by $\mathbf{Sub}(f)(H) = f^{-1}(H)$ for each $H \in \mathbf{Sub}\,G$;*
- *for each finite Gödel algebra $\mathbf{A}$, $Spec(\mathbf{A})$ is the prime spectrum of $\mathbf{A}$ and for each homomorphism of finite Gödel algebras $h : \mathbf{A} \to \mathbf{B}$, the order-preserving open map $Spec(h) : Spec(\mathbf{B}) \to Spec(\mathbf{A})$ is defined by $Spec(h)(\mathfrak{p}) = h^{-1}(\mathfrak{p})$ for each $\mathfrak{p} \in Spec(\mathbf{B})$.*

In Fig. 1, the forest $\mathscr{F}_2$ dual of the free Gödel algebra $\mathscr{G}_2$ over two variables is depicted. Nodes are marked with minterms, that is formulas corresponding to join-irreducible elements of $\mathscr{G}_2$ that generate the prime filters of $\mathscr{G}_2$.

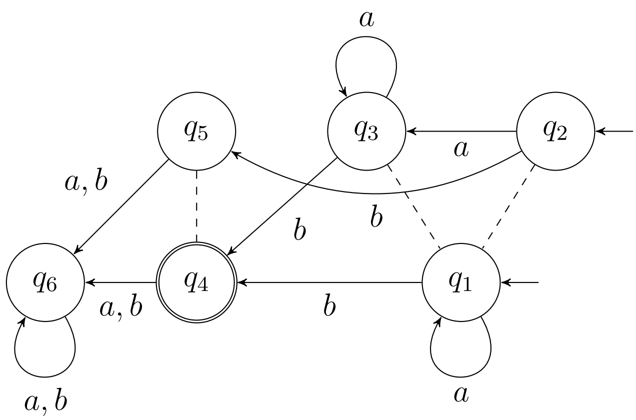


Fig. 2. Gödel automaton that 1-accepts $\mathscr{L}_1 = a^*b$ and 2-accepts $\mathscr{L}_2 = aa^*b$. Notice that $\mathscr{L}_2 \subset \mathscr{L}_1$.

3. Inheritance from the Boolean case

In this section, we briefly recall the main concepts in the paper [13] in order to pave the way to the generalisation of those concepts from classical Boolean logic to Gödel logic. For any background notion on algebraic logic we refer to [6,19].

We recall that, given a finite alphabet Σ, a *classical fortress* over $Var(n)$ is a triple $\mathcal{F} = (\varphi, \{\sigma_a : a \in \Sigma\}, \Theta)$, where φ is a formula in $Form(n)$, $\Theta \subseteq Form(n)$ is a prime theory, and each $\sigma_a : Var(n) \to Form(n)$ is a substitution mapping each variable x_i to a formula $\psi_i \in Form(n)$. Clearly, any substitution $\sigma : Var(n) \to Form(n)$ naturally extends to a map $\bar{\sigma} : Form(n) \to Form(n)$. By definition, the fortress $\mathcal{F}$ *recognises* a string $s = a_1 \cdots a_u$ if

$$\Theta \vDash \varphi[\sigma_{a_1} \circ \cdots \circ \sigma_{a_u}],$$

where $\psi[\sigma] = \bar{\sigma}(\psi)$ denotes the formula obtained by application of the substitution σ to the variables in ψ, and $\sigma_1 \circ \sigma_2$ denotes the substitution obtained by functionally composing σ_1 with σ_2.

Let us denote by $\mathscr{L}(\mathcal{F})$ the language recognised by $\mathcal{F}$. In [13] it is proved that each fortress $\mathcal{F}$ over $Var(n)$ determines a complete deterministic finite state automaton with 2^n states $\mathscr{A}(\mathcal{F})$ recognising $\mathscr{L}(\mathcal{F})$, and, by converse, each complete deterministic finite state automaton with 2^n states $\mathscr{A}$ determines a fortress $\mathcal{F}(\mathscr{A})$ over $Var(n)$ recognising the same language of $\mathscr{A}$. Since the constraint on the number of states poses no limitation on the recognising power of DFA's (see Section 6.2 for a discussion), the languages recognised by classical fortresses are exactly the regular languages.

Notice that the notion of language recognised by a fortress involves *only* the purely logical concepts of logical consequence of a formula from a (prime) theory and of substitution of variables with formulas: this is all that is needed to replicate the behaviour of every deterministic finite state automaton.

The proof that classical fortresses recognise exactly the regular languages makes use of algebraic semantics and categorical dualities, as we are going to sketch:

First, as is well known, the Lindenbaum-Tarski algebraic semantics of classical propositional logic is given by the variety of Boolean algebras. In particular, a formula $\varphi \in Form(n)$ is a logical consequence of a theory $\Theta \subseteq Form(n)$ if and only if the element $[\varphi]$ is the top element 1 of the Lindenbaum algebra given by $\mathscr{B}_n/\bar{\Theta}$, where $\mathscr{B}_n$ is the free n-generated Boolean algebra, and $\bar{\Theta}$ is the congruence generated by $\{[\psi] = 1 : \psi \in \Theta\}$. Equivalently, $[\varphi]$ belongs to the filter $[\Theta]$ of $\mathscr{B}_n$, where $[\Theta] = \{[\psi] : \psi \in \Theta\}$. Further, each substitution $\sigma : Var(n) \to Form(n)$ corresponds bijectively to an endomorphism $\tilde{\sigma} : \mathscr{B}_n \to \mathscr{B}_n$, as $\tilde{\sigma}([\psi(x_1, \ldots, x_n)]) = [\psi(\sigma(x_1), \ldots, \sigma(x_n))]$.

Second, the finite slice of Stone's categorical duality between Boolean algebras and Stone spaces proves that the category of finite Boolean algebras with homomorphisms is categorically equivalent to the opposite of the category of finite sets and maps between them. In particular, the free Boolean algebra $\mathscr{B}_n$ corresponds dually to the set of its 2^n many atoms, or, equivalently, the set $M(n)$ of maximal filters, each one of them singly generated by one of the atoms, while each (equivalence class of a) formula φ corresponds to the set of maximal filters containing $[\varphi]$, as well as each theory Θ corresponds to the set of maximal filters whose union is $[\Theta]$. Further, each endomorphism $\tilde{\sigma} : \mathscr{B}_n \to \mathscr{B}_n$ corresponds dually to a map $M(n) \to M(n)$.

The algebraic semantics given by finite Boolean algebras and the categorical duality with finite sets allows to have a dictionary to translate the definition specifying a complete deterministic finite state automaton with 2^n states into the definition of a fortress, and *viceversa*: the set of states is $M(n)$, the subset of final states corresponds to a formula φ, the initial state to a *prime* theory Θ, and the transition function to the collection of substitutions $\{\sigma_a : a \in \Sigma\}$.

In this paper we shall adapt the notion of classical fortress to Gödel infinitely valued logic. That is, a Gödel fortress will actually be a classical fortress, but the notion of acceptance is changed into

$$\Theta' \vDash_G \varphi[\sigma_{a_1} \circ \cdots \circ \sigma_{a_u}],$$

where here with $\Gamma \vDash_G \psi$ we mean that ψ is consequence in Gödel logic of the theory Γ, and Θ' is a prime theory extending the prime theory Θ.

We shall prove, using the algebraic semantics given by Gödel algebras and the categorical dual equivalence with forests, that Gödel fortresses recognises exactly the finite chains of inclusions of regular languages.

4. Gödel automata and Gödel fortresses

In this section, we define automata and fortresses based on Gödel logic and we investigate the class of languages that they recognise. We recall here the notions and corresponding notation that we are going to use:

- Formulas $\varphi \in Form(n)$ of Gödel logic;
- The free Gödel algebra $\mathscr{G}_n$ whose elements are equivalence classes $[\varphi]$ of Gödel formulas;
- The finite forest $\mathscr{F}_n$ dual to the free Gödel algebra $\mathscr{G}_n$, whose elements are the prime filters $\langle [m] \rangle$ of $\mathscr{G}_n$ generated by join-irreducible elements $[m]$ of $\mathscr{G}_n$, that is, by equivalence classes of minterms m of Gödel logic.
- Theories Θ of Gödel formulas and corresponding filters, denoted $[\Theta]$, where $[\Theta] = \{[\varphi] \mid \varphi \in \Theta\}$ of $\mathscr{G}_n$.

Definition 3 (Gödel Automaton). A *Gödel automaton* on an alphabet Σ is a quadruple $\mathscr{A} = (Q, \delta, C, F)$ such that:

1. Q is a finite forest $\mathscr{F}$;
2. $\delta : Q \times \Sigma \to Q$ is such that, for each $a \in \Sigma$, the map $\tau_a : Q \to Q$ defined by $\tau_a(q) = \delta(q, a)$ is open and order preserving over $\mathscr{F}$;
3. $C \subseteq Q$ is a downward-closed chain of the forest $\mathscr{F}$, the *chain of initial states*;
4. $F \subseteq Q$ is a downset of $\mathscr{F}$, the *downset of final states*.

If $\mathscr{F}$ is isomorphic with $\mathscr{F}_n$, then we say that $\mathscr{A}$ is a *free Gödel automaton* (over n variables).

A Gödel automaton can accept words with different levels of acceptance, depending on the considered starting state in the chain of initial states.

Definition 4. Given a Gödel automaton $\mathscr{A} = (Q, \delta, C, F)$ on an alphabet Σ, where $C = q_1 < \cdots < q_h$, we say that a word $a_1 \cdots a_u$ is *i-accepted*, for $i = 1, \ldots, h$, if

$$\tau_{a_u}(\tau_{a_{u-1}}(\cdots(\tau_{a_1}(q_i))\cdots)) \in F ,$$

where $\tau_{a_j}(q) = \delta(q, a_j)$ for every $q \in Q$, $j = 1, \ldots, u$. We set $\mathscr{L}_i = \{s \in \Sigma^* \mid s \text{ is i-accepted by } \mathscr{A}\}$.

Lemma 4. *Given a Gödel automaton $\mathscr{A} = (Q, \delta, C, F)$ as in the above definition, if $q_i < q_j$ then $\mathscr{L}_j \subseteq \mathscr{L}_i$.*

Proof. Let $q_i < q_j$ and consider $a \in \Sigma$. Since τ_a is order preserving, then $\tau_a(q_i) \le \tau_a(q_j)$. If $\tau_a(q_j) \in F$, since F is a downset then also $\tau_a(q_i) \in F$. By iterating this process, we get that for every word $a_1 \cdots a_u \in \Sigma^*$ if $\tau_{a_u}(\tau_{a_{u-1}}(\cdots(\tau_{a_1}(q_j))\cdots)) \in F$, then also $\tau_{a_u}(\tau_{a_{u-1}}(\cdots(\tau_{a_1}(q_i))\cdots)) \in F$. Hence $\mathscr{L}_j \subseteq \mathscr{L}_i$. □

The *chain of languages* accepted by a Gödel automaton $\mathscr{A} = (Q, \delta, C, F)$, where $C = q_1 < \cdots < q_h$, is

$$\mathscr{L}_h \subseteq \cdots \subseteq \mathscr{L}_1$$

where $\mathscr{L}_i = \{s \in \Sigma^* \mid s \text{ is i-accepted by } \mathscr{A}\}$.

Lemma 5. *Each language in the chain of languages accepted by a Gödel automaton is regular.*

Proof. Let $\mathscr{A} = (Q, \delta, C, F)$ be a Gödel automaton recognising the chain of languages $\mathscr{L}_h \subseteq \cdots \subseteq \mathscr{L}_1$. Let us consider q_i in C, and let Q_i be the set of states which are reachable from q_i. Let $\delta_i : Q_i \times \Sigma \to Q_i$ be the restriction of δ to the set Q_i. Let $\tau_a : Q_i \to Q_i$ be defined by $\tau_a(q) = \delta(q, a)$ for each $a \in \Sigma$. Then, by Definition 4, the automaton $(Q_i, \delta_i, q_i, F \cap Q_i)$ is a DFA recognising $\mathscr{L}_i$, hence $\mathscr{L}_i$ is regular. □

Example 3. In Fig. 2 we see an example of a Gödel automaton $\mathscr{A} = (Q, \delta, C, F)$ on the alphabet $\Sigma = \{a, b\}$.

The set $Q = \{q_i \mid i = 1, \ldots, 6\}$ equipped with the order relation given by the dashed lines is a downset of the forest $\mathscr{F}_2$ (see Fig. 1). The open and order-preserving transition function is represented with labelled arrows, $C = \{q_1, q_2\}$ and $F = \{q_4\}$.

Such automaton 1-accepts $\mathscr{L}_1 = a^*b$ and 2-accepts $\mathscr{L}_2 = aa^*b$, as, for example, $\tau_b(q_1) = q_4 \in F$, but $\tau_b(q_2) = q_5 \notin F$. Moreover, $\mathscr{L}_1 = \mathscr{L}(\mathscr{A}_1)$ and $\mathscr{L}_2 = \mathscr{L}(\mathscr{A}_2)$ with $\mathscr{A}_1 = (Q_1, \delta_1, q_1, F)$, $\mathscr{A}_2 = (Q_2, \delta_2, q_2, F)$, where $Q_1 = \{q_1, q_4, q_6\}$ and $Q_2 = \{q_i\}_{i=2}^6$ are the set of states that are reachable from q_1 and q_2, and δ_1, δ_2 are the restrictions of δ to Q_1 and Q_2 respectively. The regular languages $\mathscr{L}_1$ and $\mathscr{L}_2$ satisfy the inclusion $\mathscr{L}_2 \subset \mathscr{L}_1$.

Definition 5 (Gödel Fortress). A Gödel fortress in n variables over an alphabet Σ is a triple $\mathcal{F} = (\varphi, \{\sigma_a\}_{a\in\Sigma}, \Theta)$, where

1. φ is a Gödel formula with variables in $Var(n)$;
2. for each $a \in \Sigma$, σ_a is a substitution, i.e. $\sigma_a : Var(n) \to Form(n)$;
3. Θ is a prime theory in the variables $Var(n)$.

Definition 6. Given the fortress $\mathcal{F} = (\varphi, \{\sigma_a\}_{a\in\Sigma}, \Theta)$, let

$$\Theta_1 \supset \Theta_2 \supset \cdots \supset \Theta_h = \Theta$$

be the maximal chain of inclusions of prime theories including Θ (see Lemma 3). A word $s = a_1 \cdots a_u$ in Σ^* is i-accepted by $\mathcal{F}$ if

$$\Theta_i \vDash \varphi[\sigma_{a_1} \circ \cdots \circ \sigma_{a_u}]$$

that is, the formula obtained from φ by substituting variables via the composite substitution $\sigma_{a_1} \circ \cdots \circ \sigma_{a_u}$ is a logical consequence of the theory Θ_i. We define $\mathscr{L}_i$ as the set of words i-accepted by $\mathcal{F}$.

Let us notice that if $i < j$ then $\Theta_j \subset \Theta_i$, and then if $\Theta_j \models \varphi[\sigma_{a_1} \circ \cdots \circ \sigma_{a_u}]$ also $\Theta_i \models \varphi[\sigma_{a_1} \circ \cdots \circ \sigma_{a_u}]$. Hence $\mathscr{L}_j \subseteq \mathscr{L}_i$. We conclude that $\mathscr{L}_h \subseteq \cdots \subseteq \mathscr{L}_1$.

The proof of the following proposition is a straightforward consequence of the definition of i-acceptance:

Proposition 2. *Let φ and ψ be two logically equivalent formulas and, for every $a \in \Sigma$ and $x \in Var(n)$, let $\sigma_a(x)$ be logically equivalent to $\sigma'_a(x)$. Then the Gödel fortresses $(\varphi, \{\sigma_a\}_{a\in\Sigma}, \Theta)$ and $(\psi, \{\sigma'_a\}_{a\in\Sigma}, \Theta)$ accept exactly the same chain of languages.*

Using Theorem 1 we can show a correspondence between fortresses and Gödel automata.

Recalling that the forest $\mathscr{F}_n = Spec(\mathscr{G}_n)$ is made by the prime filters of $\mathscr{G}_n$ ordered by reverse inclusion, it is easy to check that for every formula φ the set

$$\{\mathfrak{p} \in Spec(\mathscr{G}_n) \mid [\varphi] \in \mathfrak{p}\}$$

is a downset of $\mathscr{F}_n$. With this notation we have, for any theory Θ:

$$\Theta \models \varphi \text{ iff } \varphi \in \Theta \text{ iff } [\Theta] \in \{\mathfrak{p} \in Spec(\mathscr{G}_n) \mid [\varphi] \in \mathfrak{p}\}\,.$$

We recall that every substitution $\sigma : Var(n) \to Form(n)$ can be naturally extended to a homomorphism $\widetilde{\sigma}$ from $\mathscr{G}_n$ to itself, where $\widetilde{\sigma}([\varphi]) = [\varphi[\sigma]]$. When considering its dual counterpart, we get the open and order-preserving map $Spec(\widetilde{\sigma}) : \mathfrak{p} \in \mathscr{F}_n \mapsto \widetilde{\sigma}_a^{-1}(\mathfrak{p}) \in \mathscr{F}_n$.

Definition 7 (From fortress to automaton)**.** Given a fortress

$$\mathcal{F} = (\varphi, \{\sigma_a\}_{a\in\Sigma}, \Theta)$$

in n variables over Σ, we consider the free Gödel automaton $\mathscr{A}_{\mathcal{F}} = (Q, \delta, C, F)$ over Σ, where:

1. $Q = \mathscr{F}_n$, is the finite forest dual to the free Gödel algebra $\mathscr{G}_n$,
2. $\delta : Q \times \Sigma \to Q$ is defined by $\delta(\mathfrak{p}, a) = \tau_a(\mathfrak{p}) = \widetilde{\sigma}_a^{\,-1}(\mathfrak{p})$,
3. $C = \{[\Theta] = \mathfrak{p}_h \subseteq \cdots \subseteq \mathfrak{p}_1\}$ is the chain of prime filters containing $[\Theta]$,
4. $F = \{\mathfrak{p} \in Spec(\mathscr{G}_n) \mid [\varphi] \in \mathfrak{p}\}$ is the downset of $\mathscr{F}_n$ determined by φ.

Theorem 2. *The Gödel fortress $\mathcal{F}$ and the Gödel automaton $\mathscr{A}_{\mathcal{F}}$ recognise the same chain of languages.*

Proof. Let $\mathcal{F} = (\varphi, \{\sigma_a\}_{a\in\Sigma}, \Theta)$ be a Gödel fortress in n variables such that

$$\Theta = \Theta_h \subseteq \cdots \subseteq \Theta_1$$

is the maximal chain of inclusions of prime filters including Θ and let $\mathscr{A}_{\mathcal{F}}$ as in Definition 7. Let $s = a_1 \cdots a_u$. Then s is i-accepted by $\mathcal{F}$ if and only if $\Theta_i \models \varphi[\sigma_{a_1} \circ \cdots \circ \sigma_{a_u}]$. Let us write $\sigma = \sigma_{a_1} \circ \cdots \circ \sigma_{a_u}$, then $\Theta_i \models \sigma[\varphi]$, hence $\sigma[\varphi] \in \Theta_i$. In order to show that $\mathscr{A}_{\mathcal{F}}$ i-accepts s, we have to prove that $\widetilde{\sigma}_{a_u}^{-1} \circ \cdots \circ \widetilde{\sigma}_{a_1}^{-1}([\Theta_i]) \in \{\mathfrak{p} \in Spec(\mathscr{G}_n) \mid [\varphi] \in \mathfrak{p}\}$, hence that $[\varphi] \in \widetilde{\sigma}_{a_u}^{-1} \circ \cdots \circ \widetilde{\sigma}_{a_1}^{-1}([\Theta_i])$. And indeed:

$$\sigma[\varphi] \in \Theta_i \text{ iff } \widetilde{\sigma}([\varphi]) \in [\Theta_i] \text{ iff } [\varphi] \in \widetilde{\sigma}^{-1}([\Theta_i]) = (\widetilde{\sigma}_{a_1} \circ \cdots \circ \widetilde{\sigma}_{a_u})^{-1}([\Theta_i])\,.$$

We hence proved that s is i-accepted by $\mathcal{F}$ if and only if it is i-accepted by $\mathscr{A}_{\mathcal{F}}$. □

For each $\varphi \in Form(n)$ we let M_φ be the set of minterms $\{m \mid [m] \leq [\varphi]\}$.

Definition 8 (From automaton to fortress)**.** Given a free Gödel automaton over n variables $\mathscr{A} = (\mathscr{F}_n, \delta, C, F)$ over an alphabet Σ, we consider the fortress $\mathcal{F}_{\mathscr{A}} = (\varphi^F, \{\sigma_a\}_{a\in\Sigma}, \Theta^C)$ over Σ where:

- $\varphi^F = \bigvee_{m\in M} m$ where $M = \{m \mid \langle [m]\rangle \in F\}$;
- displaying C as $[\Theta_h] \subset \cdots \subset [\Theta_1]$, let $\Theta^C = \Theta_h$;
- for every $a \in \Sigma$ and minterm $m \in Form(n)$ let

$$\sigma_a(m) = \bigvee \{\, t \mid \langle [t]\rangle \in \tau_a^{-1}(\langle [m]\rangle)\,\}\,.$$

Notice that the definition of σ_a on minterms determines the definition of σ_a as a substitution, since, for each $x \in Var(n)$ we have that $\sigma_a(x) = \sigma_a(\bigvee_{m\in M_x} m)$, that is,

$$\sigma_a(x) = \bigvee \{\, t \mid \langle [t]\rangle \in \tau_a^{-1}(\langle [m]\rangle), m \in M_x\,\}\,.$$

We further extend $\widetilde{\sigma}_a : \mathscr{G}_n \to \mathscr{G}_n$ to arbitrary downward-closed subsets S of $\mathscr{F}_n$ by putting $\widetilde{\sigma}_a(S) = \{\langle \widetilde{\sigma}_a([\varphi])\rangle \mid \langle [\varphi]\rangle \in S\}$.

Theorem 3. *The chain of languages recognised by a free Gödel automaton $\mathscr{A}$ is the same chain of languages recognised by the Gödel fortress $\mathcal{F}_{\mathscr{A}}$.*

Proof. Let $\mathscr{A} = (\mathscr{F}_n, \delta, C, F)$ be a free Gödel automaton with $C = [\Theta_h] \subset \cdots \subset [\Theta_1]$. The word $s = a_1 \cdots a_u$ is i-accepted by $\mathscr{A}$ if and only if

$$(\tau_{a_u} \circ \cdots \circ \tau_{a_1})([\Theta_i]) \in F\,.$$

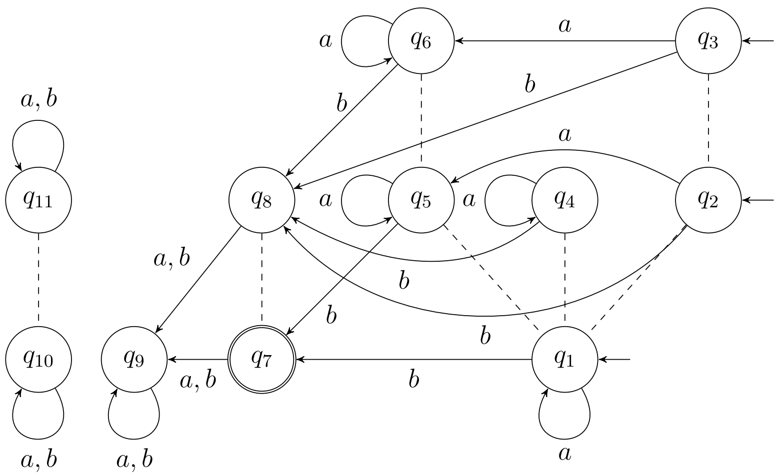


Fig. 3. Free Gödel automaton accepting $\emptyset \subset aa^*b \subset a^*b$.

Now, $\mathcal{F}_{\mathscr{A}}$ i-accepts s iff $\Theta_i \models \varphi^F[\sigma_{a_1} \circ \cdots \circ \sigma_{a_u}]$ iff $[\Theta_i] \in \{\mathfrak{p} \in \mathscr{F}_n \mid [\,\varphi^F[\sigma_{a_1} \circ \cdots \circ \sigma_{a_u}]\,] \in \mathfrak{p}\}$ iff $\sigma^{-1}_{a_1}([\Theta_i]) \in \{\mathfrak{p} \in \mathscr{F}_n \mid [\,\varphi^F[\sigma_{a_2} \circ \cdots \circ \sigma_{a_u}]\,] \in \mathfrak{p}\}$ iff $(\sigma^{-1}_{a_u} \circ \cdots \circ \sigma^{-1}_{a_1})([\Theta_i]) \in \{\mathfrak{p} \in \mathscr{F}_n \mid [\,\varphi^F\,] \in \mathfrak{p}\}$. But $\{\mathfrak{p} \in \mathscr{F}_n \mid [\,\varphi^F\,] \in \mathfrak{p}\} = F$ by the definition of φ^F. Whence, $\mathcal{F}_{\mathscr{A}}$ i-accepts s iff

$$(\sigma^{-1}_{a_u} \circ \cdots \circ \sigma^{-1}_{a_1})([\Theta_i]) \in F\,.$$

We observe that for each $a \in \Sigma$ and each minterm m,

$$\widetilde{\sigma}_a(\{\langle[m]\rangle\}) = \{\,\langle[t]\rangle \mid \langle[t]\rangle \in \tau^{-1}_a(\langle[m]\rangle)\,\} = \tau^{-1}_a(\langle[m]\rangle)\,.$$

Whence, $(\sigma^{-1}_{a_u} \circ \cdots \circ \sigma^{-1}_{a_1})([\Theta_i]) \in F$ iff $[\Theta_i] \in (\widetilde{\sigma}_{a_1} \circ \cdots \circ \widetilde{\sigma}_{a_u})(F)$ iff $[\Theta_i] \in (\tau^{-1}_{a_1} \circ \cdots \circ \tau^{-1}_{a_u})(F)$ iff $(\tau_{a_u} \circ \cdots \circ \tau_{a_1})([\Theta_i]) \in F$. That is to say that $\mathcal{F}_{\mathscr{A}}$ i-accepts s iff $\mathscr{A}$ i-accepts s. □

4.1. An example

Consider the forest $\mathscr{F}_2$, dual of the free Gödel algebra over two variables $\mathscr{G}_2$. In Fig. 1, nodes of the forest $\mathscr{F}_2$ are labelled by the minterms m_i whose equivalence classes $[m_i]$ generates all the prime filters $q_i = \langle[m_i]\rangle$ of $\mathscr{G}_2$. On $\mathscr{F}_2$ we consider the Gödel automaton $\mathscr{A} = (\mathscr{F}_2, \delta, C, F)$ in Fig. 3, where $F = \{q_7\}$ is the filter generated by $m_7 = y \wedge \neg x$ and C is the chain $q_1 < q_2 < q_3$ corresponding to filters generated by minterms $m_1 = x \wedge y$, $m_2 = x \wedge \neg\neg y$ and $m_3 = \neg\neg y \wedge ((x \to y) \to x)$. Since the chain C has three elements, $\mathscr{A}$ accepts a chain of three languages. It can be checked that the accepted chain of languages is $\emptyset \subset aa^*b \subset a^*b$.

We are going to build the fortress $\mathcal{F}_{\mathscr{A}}$ as in Definition 8, using the two variables x and y.

- Since $\{m \mid \langle[m]\rangle \in F\} = \{m_7\}$ then $\varphi^F = m_7$.
- The prime theory Θ^C is such that $[\Theta^C] = q_3$ hence we can take $\Theta^C = \{\psi \mid m_3 \models \psi\}$.
- In order to describe σ_a and σ_b, we focus on variables. From Fig. 1, we can check that the minterms generating filters containing x (i.e., minterms that are *smaller* than x) are those marked with white circles, namely $M_x = \{m_{10}, m_1, m_2\}$, while those corresponding to the variable y are the squared ones, namely $M_y = \{m_7, m_1, m_5\}$. Checking τ^{-1}_a and τ^{-1}_b in $\mathscr{A}$, we get:

$$\tau^{-1}_a(\langle[m_1]\rangle) = \{q_1\}; \qquad \tau^{-1}_b(\langle[m_1]\rangle) = \emptyset;$$
$$\tau^{-1}_a(\langle[m_2]\rangle) = \emptyset; \qquad \tau^{-1}_b(\langle[m_2]\rangle) = \emptyset;$$
$$\tau^{-1}_a(\langle[m_{10}]\rangle) = \{q_{10}\}; \qquad \tau^{-1}_b(\langle[m_{10}]\rangle) = \{q_{10}\};$$
$$\tau^{-1}_a(\langle[m_5]\rangle) = \{q_2, q_5\}; \qquad \tau^{-1}_b(\langle[m_5]\rangle) = \emptyset;$$
$$\tau^{-1}_a(\langle[m_7]\rangle) = \emptyset; \qquad \tau^{-1}_b(\langle[m_7]\rangle) = \{q_1, q_5\}.$$

Hence

$$\sigma_a(x) = m_1 \vee m_{10} \qquad \sigma_b(x) = m_{10}$$
$$\sigma_a(y) = m_1 \vee m_2 \vee m_5 \equiv m_2 \vee m_5 \qquad \sigma_b(y) = m_1 \vee m_5 \equiv m_5.$$

Consider now the string $s = ab$. In order to speed up the calculations with substitutions, we refer to the syntax of minterms as in Fig. 1 and we note that the negation $\neg m$ of a minterm is the disjunction of minterms on the trees not containing m. Further, the double negation $\neg\neg m$ is the disjunction of all minterms in the same tree as m, cfr. Theorem 1. We start from $\varphi^F = m_7 = y \wedge \neg x$ and apply substitutions σ_a and σ_b on both variables:

$$\varphi^F[\sigma_b] = m_5 \wedge \neg m_{10} \equiv m_5 = y \wedge \neg\neg x$$

(since $\neg m_{10} = \bigvee_{i=1}^{9} m_i$) and

$$(\varphi^F[\sigma_b])[\sigma_a] = (m_2 \vee m_5) \wedge \neg\neg(m_1 \vee m_{10}) = m_2 \vee m_5,$$

(since $\neg\neg(m_1 \vee m_{10}) = m_3 \vee m_4 \vee m_6 \vee m_{11}$). We have $m_2 \vDash m_2 \vee m_5$ and $m_1 \vDash m_2 \vee m_5$, while $m_3 \nvDash m_2 \vee m_5$. Hence, the string ab is 1-accepted and 2-accepted, but it is not 3-accepted.

5. From a chain of languages to a free Gödel automaton

Now that we have already exposed the correspondence between a fortress and a (free) Gödel automaton, we show a procedure to get a (free) Gödel automaton from a chain of languages.

Theorem 4. *Given a chain of regular languages*

$$\mathscr{L}_h \subseteq \cdots \subseteq \mathscr{L}_1$$

over Σ, there is a Gödel automaton $\mathscr{A} = (Q, \delta, C, F)$ with $|C| = h$ that accepts $\mathscr{L}_h \subseteq \cdots \subseteq \mathscr{L}_1$.

Proof. We split the proof into subtasks:

Construction of classical automata: for each $\mathscr{L}_j$ in the chain, we fix a complete deterministic finite-state automaton $\mathscr{A}(\mathscr{L}_j) = (Q_j, \delta_j, i_j, F_j)$ that recognises it.

Intersection: we let $\mathscr{A}_1 = \mathscr{A}(\mathscr{L}_1)$ and, for $j \in [2, h]$, $\mathscr{A}_j = \mathscr{A}_{j-1} \cap \mathscr{A}(\mathscr{L}_j)$, *i.e.*, the intersection automaton of $\mathscr{A}(\mathscr{L}_j)$ and the one resulting from the preceding iteration. At the end of this stage, we have the set of automata $\{\mathscr{A}_1, \dots, \mathscr{A}_h\}$ each one of them denoted by $\mathscr{A}_j = (Q_j^\cap, \delta_j^\cap, i_j^\cap, F_j^\cap)$.

For each $k \in [1, h]$, every state of $\mathscr{A}_k$ is a nested tuple of the form $(\cdots(((q_1, q_2), q_3), \dots, q_k)$, where each q_j is a state of the automaton $\mathscr{A}(\mathscr{L}_j)$. To simplify notation we write $(q_1 q_2 q_3 \cdots q_k)$ instead of $(\cdots(((q_1, q_2), q_3) \cdots q_k)$.

Construction of the Gödel automaton: we build the automaton $\mathscr{A} = (Q, \delta, C, F)$ as follows.

Let $Q = \bigcup_{k=1}^{h} Q_k^\cap$ be the disjoint union of the set of all states of $\mathscr{A}_1, \dots, \mathscr{A}_h$. We endow Q with the following relation: given $p, r \in Q$, we stipulate $p \le r$ iff p is a prefix of r, that is, $p = (p_1 p_2 \cdots p_v)$ and $r = (r_1 r_2 \cdots r_w)$, with $v \le w$ and $p_j = r_j$ for each $j \in [1, v]$. It is straightforward to check that $\le$ is a partial order relation on Q such that $(Q, \le)$ is a forest in which the states of $\mathscr{A}_1$ are the roots.

We consider the initial state $(i_1 i_2 \cdots i_h)$ for each i_j being the initial state of $\mathscr{A}(\mathscr{L}_j)$. Notice that each prefix $(i_1 i_2 \cdots i_k)$ (for $k \in [1, h]$) is the initial state of $\mathscr{A}_k$. Clearly, the sequence C of initial states of $\mathscr{A}_1, \dots, \mathscr{A}_h$, that is $C = (i_1), (i_1 i_2), \cdots, (i_1 i_2 \cdots i_h)$ is a downward-closed chain in Q.

Further, we take the set $F = \bigcup_{k=1}^{h} F_k^\cap$ as the set of final states, where each $F_k^\cap$ is the set of final states of $\mathscr{A}_k$. In order to prove that F is a downset of Q, note that if $r \in F$ then $r \in F_k^\cap$ for some $k \in [1, h]$, that is, $r = (r_1 \cdots r_k)$ where each $r_j \in Q_j$. If $p \le r$ then $p = (r_1 \cdots r_v)$ for some $v \le k$, and by Definition 2 of intersection of automata, $p \in F_v^\cap$, whence $p \in F$.

In order to define the transition function $\delta : Q \times \Sigma \to Q$, for each state $(p_1 \cdots p_v) \in Q$ and each $a \in \Sigma$ we set

$$\delta((p_1 \cdots p_v), a) = (r_1 \cdots r_v),$$

where, for each $j \in [1, v]$, $\delta_j(p_j, a) = r_j$.

We need to prove that $\delta(\_, a)$ is an open and order-preserving map of the forest $(Q, \le)$.

Indeed, if $p \le r$, then $r = (r_1 \cdots r_w)$ and $p = (r_1 \cdots r_v)$ for $v \le w$. Whence, for suitable states $t_1, \dots, t_w$, where each $t_j \in Q_j$, we have that $\delta(r, a) = \delta((r_1 \cdots r_w), a) = (t_1 \cdots t_w)$, while $\delta(p, a) = \delta((r_1 \cdots r_v), a) = (t_1 \cdots t_v)$, that is $\delta(p, a) \le \delta(r, a)$.

In order to prove the openness property of δ, let $t \le \delta(r, a)$ for some states $r, t \in Q$, and symbol $a \in \Sigma$. Then $t = (t_1 \cdots t_v)$ and $\delta(r, a) = (t_1 \cdots t_w)$ for suitable states $t_1, \dots, t_w$, each $t_j \in Q_j$ and $v \le w$. Then, there are states $r_1, \dots, r_w$, with each $r_j \in Q_j$, such that $r = (r_1 \cdots r_w)$ (that is, $\delta_j(r_j, a) = t_j$). But then $p = (r_1 \cdots r_v)$ is a state of Q such that $p \le r$ and $\delta(p, a) = (t_1 \cdots t_v) = t$.

Finally, we check that the chain of languages accepted by $\mathscr{A}$ is exactly as desired: $\mathscr{L}_h \subseteq \cdots \subseteq \mathscr{L}_1$. Indeed, by Proposition 1, $\mathscr{L}(\mathscr{A}_k) = \mathscr{L}(\mathscr{A}_{k-1}) \cap \mathscr{L}(\mathscr{A}(\mathscr{L}_k))$ and since $\mathscr{L}(\mathscr{A}(\mathscr{L}_k)) = \mathscr{L}_k$, $\mathscr{L}(\mathscr{A}_1) = \mathscr{L}_1$ and $\mathscr{L}_h \subseteq \cdots \subseteq \mathscr{L}_1$, then $\mathscr{L}(\mathscr{A}_k) = \mathscr{L}_k$ for any $k \in [1, h]$. □

Note that during the proof, it was not necessary to impose constraints on the properties of the automata $\mathscr{A}(\mathscr{L}_i)$—for example, minimality on the number of states. Therefore, given the same chain of languages, the construction may yield Gödel automata that are quite different from one another. Furthermore, the proof provides only one of the possible ways to construct the required Gödel automaton. In fact, starting from the result thus obtained, in some cases, it is possible to merge certain states into one that corresponds to their prefix, as observed in Example 4, allowing us to reduce the total number of states.

Example 4. Let us consider two languages $\mathscr{L}_1 = a^*b$ and $\mathscr{L}_2 = aa^*b$. For each of them, we can take the minimal automaton recognising the language by setting $\mathscr{A}(\mathscr{L}_1) = (\{q_1, q_2, q_3\}, \delta_1, q_1, \{q_2\})$ and $\mathscr{A}(\mathscr{L}_2) = (\{q_4, q_5, q_6, q_7\}, \delta_2, q_4, \{q_7\})$ where

δ_1	q_1	q_2	q_3
a	q_1	q_3	q_3
b	q_2	q_3	q_3

δ_2	q_4	q_5	q_6	q_7
a	q_5	q_5	q_7	q_7
b	q_7	q_6	q_7	q_7

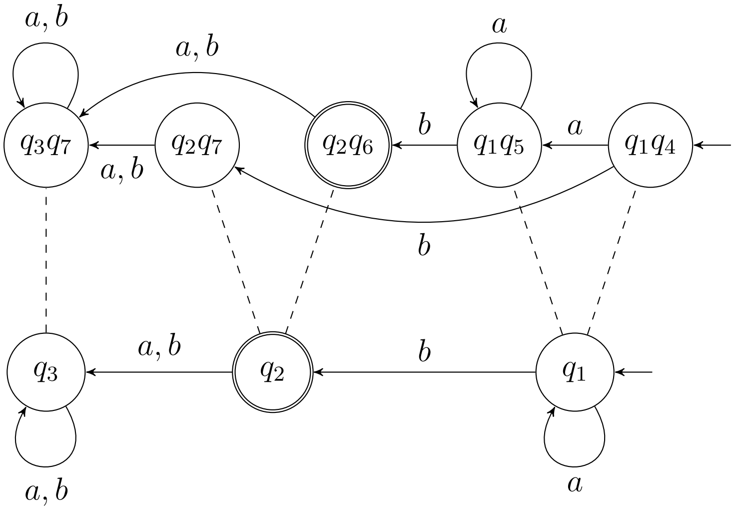


Fig. 4. Gödel automaton corresponding to the chain $aa^*b \subseteq a^*b$.

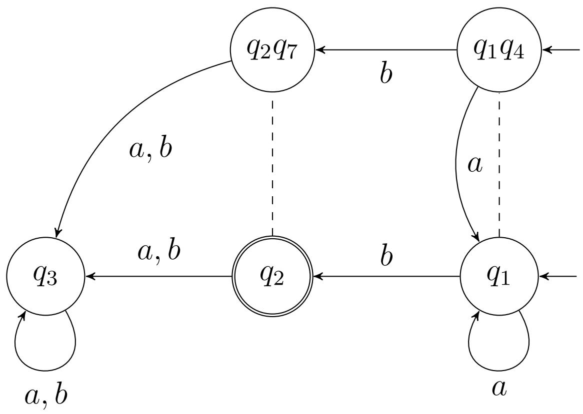


Fig. 5. Alternative Gödel automaton corresponding to the chain $aa^*b \subseteq a^*b$.

In Fig. 4, a Gödel automaton recognising the chain of languages $\mathscr{L}_2 = aa^*b \subseteq \mathscr{L}_1 = a^*b$ is depicted. The lower automaton in the picture is $\mathscr{A}(\mathscr{L}_1)$ while the upper one is the intersection of $\mathscr{A}(\mathscr{L}_1)$ and $\mathscr{A}(\mathscr{L}_2)$. States in the intersection are denoted by sequences of states of the two initial automata and the dashed line is the order relation of being a prefix. The initial states are q_1 and q_1q_4, while q_2 and q_2q_6 are final. The transiction function is inherited from $\mathscr{A}(\mathscr{L}_1)$ and $\mathscr{A}_2 = \mathscr{A}(\mathscr{L}_1) \cap \mathscr{A}(\mathscr{L}_2)$.

Notice that some states can be collapsed into their "prefix" state. As shown in Fig. 5, for the given example, q_1q_5, q_2q_6 and q_3q_7 can be collapsed in their prefixes by migrating their incoming arrows respectively to q_1, q_2 and q_3, without modifying either the fundamental properties of the transition map of the automaton or the accepted chain of languages.

We recall that for each integer $n \geq 0$, there exists a forest $\mathscr{F}_n$, unique up to isomorphism, that serves as the dual of the free n-generated Gödel algebra. Leveraging the combinatorial structure of $\mathscr{F}_n$, a classification algorithm for forests, called *Linnaeus* ([22], Algorithm 2), has been developed that classifies a given forest $\mathscr{F}$ by determining the smallest non-negative integer m such that $\mathscr{F}$ is isomorphic to a subforest of $\mathscr{F}_m$.

Theorem 5. *For each Gödel automaton there is a free Gödel automaton recognising the same chain of regular languages.*

Proof. Let $(\mathscr{F}, \delta, C, F)$ be a Gödel automaton and n be the minimum integer such that $\mathscr{F} \subseteq \mathscr{F}_n$ where $\mathscr{F}_n = Spec(\mathscr{G}_n)$. $\mathscr{F}_n$ will constitute the underlying structure of the corresponding free Gödel automaton. Let $f : \mathscr{F} \hookrightarrow \mathscr{F}_n$ be the open and order-preserving injection that maps $\mathscr{F}$ isomorphically onto a subforest of $\mathscr{F}_n$. Clearly $f(C)$ is a downward-closed chain of $\mathscr{F}_n$, as well as $f(F)$ is a downset of $\mathscr{F}_n$. The injection f maps each tree T of $\mathscr{F}$ onto a Gödel tree $f(T)$ that is a subtree of some tree T' of $\mathscr{F}_n$ having enough nodes per level to host it. Moreover, the injection f lets $f(T)$ inherit the open and order-preserving maps $\tau_a = \delta(q, a)$ of T: that is, we define $f(\tau_a) : f(\mathscr{F}) \to f(\mathscr{F})$ by $f(\tau_a)(q) = f(\tau_a(q))$ for all $q \in f(\mathscr{F})$. Notice that, the map $f(\tau_a)$ may be not completely defined over T' because f is not necessarily surjective. To extend the map $f(\tau_a)$ to the whole of $\mathscr{F}_n$ we have to consider every node $x \in \mathscr{F}_n \setminus f(\mathscr{F})$. Considering the natural order enforced by the forest $\mathscr{F}_n$, for every state x such that there is no node $y \in f(\mathscr{F})$ such that $x > y$, we add to the definition of $f(\tau_a)$ the set of transitions $x \xrightarrow{a} x$ for each $a \in \Sigma$. For all the other states x, there is a node y being the maximal one such that $y \in f(\mathscr{F})$ and $x > y$. For every such node, we add to the definition of $f(\tau_a)$ a transition $x \xrightarrow{a} \delta(y, a)$ for all $a \in \Sigma$. Clearly, the resulting map $f(\tau_a)$ is open, order-preserving, and defined on the whole of $\mathscr{F}_n$. Whence, letting $f(\delta): \mathscr{F}_n \times \Sigma \to \mathscr{F}_n$ be defined by $f(\delta)(q, a) = f(\tau_a)(q)$ for all $q \in \mathscr{F}_n$, we conclude that $(\mathscr{F}_n, f(\delta), f(C), f(F))$ is a free Gödel automaton recognising the same chain of languages as $(\mathscr{F}, \delta, C, F)$. □

The just-mentioned procedure is a way to complete an open and order-preserving map by maintaining both the openness and the order-preservation properties: this completion is far from being unique.

Theorem 6. *The class of Gödel fortresses recognises exactly the class of finite chains of regular languages.*

Proof. The proof follows straightforwardly from Theorems 2–5. □

6. Observations on fortresses, Gödel automata, and Fuzzy automata

In this section we collect some further results and observations concerning classical and Gödel fortresses and their associated classes of automata, and we investigate a relationship between Gödel automata and Fuzzy automata, as defined, for instance, in [14].

6.1. Generalising fortresses to other logics

Let L be a non-classical logic in the same language of classical propositional logic. We can then generalise the notion of classical fortress to a notion of fortress in L: we only need to use the notion of logical consequence in L instead of the classical notion. That is, an L-fortress is actually a classical fortress, but the notion of acceptance is changed into

$$\Theta \vDash_L \varphi[\sigma_{a_1} \circ \cdots \circ \sigma_{a_u}],$$

where with $\Gamma \vDash_L \psi$ we mean that ψ is consequence in the logic L of the theory Γ.

The fact that L-fortresses are exact descriptors of a class of formal languages will constitute the natural generalisation of the fact that classical fortresses are exact descriptors of the class of regular languages, contributing in this way to clarify some aspects of the natural semantics of L. The task then is to identify the class of formal languages recognised by L-fortresses. To accomplish this task, it would be extremely useful to be able to construct a new dictionary that allows us to *translate* a fortress into a computational model akin to a finite state automaton. We can try to build the correspondence *L-fortress—L-automaton* if we can carry over from the classical case some of the key concepts used there: 1) L is algebraisable, and the Lindenbaum-Tarski algebras of L constitute a locally finite variety $\mathbb{L}$; 2) There is a Stone-type categorical duality between the finite algebras in $\mathbb{L}$ and some category C sufficiently akin to the category of finite sets. As an instance of this approach, in this paper we have dealt with the case L is Gödel infinitely-valued logic.

6.2. On the size of the automata determined by fortresses

The fact that the number of states of the DFA determined by a fortress is a power of two is not a limitation, as each DFA with k states can be extended to a reachable DFA with 2^n states, for some $n \geq \log_2 k$, recognising the same language. On the other hand, [13] introduces the notion of *reduced* fortresses, which are associated with DFA's with no constraints on the number of states. This is achieved by adding to the definition of classical fortress another theory, which defines a subset of the atoms of $\mathcal{B}_n$ as the set of states of the described automa.

Analogously, the fact that a free Gödel automaton has by definition the number of states corresponding to the cardinality of the prime spectrum of a free Gödel algebra $\mathscr{G}_n$ is not a limitation, as we have shown how to embed any Gödel automaton into a free one. Furthermore, it is possible to introduce a notion of *reduced* Gödel fortress, analogous to the notion of reduced classical fortress.

For what regards the cardinalities of the set of states of a free Gödel automaton, the prime spectrum of $\mathscr{G}_n$ has exactly $4B(n) - 1$ elements, where $B(n)$ is the nth ordered Bell number (or Fubini number), which in turn counts the ordered partitions of a set with n elements (as a matter of fact, each element of $GR(n)$ corresponds to an ordered partition of the set consisting of the generators together with 0 and 1).

6.3. Prime, maximal, and general theories

We collect here some observations about classical fortresses that have not been developed in [13].

In a fortress $(\varphi, \{\sigma_a : a \in \Sigma\}, \Theta)$, the theory Θ is assumed to be a *prime* theory, that is, it corresponds to a prime filter of $\mathscr{B}_n$. One may observe that, being $\mathscr{B}_n$ a Boolean algebra, a filter is prime iff it is maximal.

Whence, one may wonder why the notion of being prime for the theory Θ has been preferred to the notion of being maximal. Actually, this choice is to maintain an approach suitable to be generalised: the paper [13] proposes some directions for generalisation, and in general, prime theories and maximal theories are distinct concepts and play a different rôle (however, we recall that every maximal theory is prime). In particular, in the algebraic semantics of non-classical logics, as for instance, in Gödel algebras, prime theories are not necessarily maximal.

The second observation in the choice of limiting to prime theories in the definition of fortresses is actually just a simplifying *computational* consideration, which is tacit in [13]; the theory Θ is prime since it mimics the unique initial state of a complete deterministic finite state automaton. If we relax, in the definition of fortress, the requirement that Θ is prime, and we allow any theory Γ to be admissible, nothing changes in the set of recognised languages: as a matter of fact, no extra language is recognised, since, in the automaton view, Γ describes a set of initial states over the same automaton, and a word is then accepted iff it is accepted starting from any one of the states in Γ. Whence, $\Gamma \vDash_L \varphi[\sigma_{a_1} \circ \cdots \circ \sigma_{a_u}]$ iff $\Theta \vDash_L \varphi[\sigma_{a_1} \circ \cdots \circ \sigma_{a_u}]$ for each prime theory Θ extending Γ.

The same considerations on theories apply to Gödel fortresses. As a matter of fact, in Gödel algebras, prime theories are not necessarily maximal. Moreover, if we replace in the definition of fortress the prime theory with a general theory, we do not change the recognising power of Gödel fortresses, for the same reasons expounded above for the case of classical fortresses.

6.4. On the Büchi-Elgot-Trakhtenbrot theorem

It is worthwhile stressing that fortresses are not meant to *describe*, using the language of logic, the *dynamics* of an automaton (as this, by Büchi-Elgot-Trakhtenbrot Theorem, [23–25] can be done in the monadic fragment of classical second-order logic (see below for details), but, instead they are meant to *mimic* that dynamics through fundamental concepts of logic. Of course, once the correspondence *fortress-automaton* is established, one can, in principle, use only the formal notion of fortress and the logic concepts of consequence and substitution to determine whether any given string s belongs to the language recognised by the fortress.

We now state a straightforward corollary of the correspondence *classical fortress-regular language* which could be of independent interest.

We recall that a finite word $w = a_1 a_2 \cdots a_u \in \Sigma^*$ can be conceived as a structure $\mathcal{W}(w) = (W, \leq, \{\bar{a}\}_{a\in\Sigma})$, where the universe W is a set of cardinality u, $\leq$ is a total order induced on W and for each $a \in \Sigma$, $\bar{a}$ is a one-place predicate such that for all $x \in W$, there is at most one $a \in \Sigma$ such that $\bar{a}(x)$ holds (if x is the ith element of $(W, \leq)$ then $\bar{a}_i(x)$ holds). Then a *language of finite words* L over Σ is *monadic definable* if there is a second-order monadic sentence φ such that $L = \{w : \mathcal{W}(w) \vDash \varphi\}$.

Büchi-Elgot-Trakhtenbrot Theorem states that a language of finite words is monadic definable iff it is regular. Then, the following result is straightforward.

Corollary 1. *A language of finite words is monadic definable iff it is recognised by a classical fortress.*

Since all passages in Büchi-Elgot-Trakhtenbrot Theorem are effective, as well as the passage from regular language to classical fortress, we can reduce satisfiability of second-order monadic sentences over finite words to the decidable problem of logical consequence (from a finite theory) in classical propositional logic.

6.5. Chains of languages as fuzzy languages

Let us denote with $\mathbf{L} = (L, \leq)$ a complete lattice with support set L and lattice order $\leq$. Recall that there is a uniquely determined Gödel algebra $\mathbf{G}$ having $\mathbf{L}$ as its lattice reduct. We consider a *fuzzy language* $\mathcal{L}$ over an alphabet Σ as an L-fuzzy subset of Σ^*, that is, a function $\mathcal{L} : \Sigma^* \to L$. For any $s \in \Sigma^*$, let $\mathcal{L}(s)$ be the degree of membership of s in the language $\mathcal{L}$. Recall that for each $\alpha \in L$, the α-cut of a fuzzy set $\mathcal{L}$ in Σ^* is the subset $\{s \in \Sigma^* \mid \mathcal{L}(s) \geq \alpha\} \subseteq \Sigma^*$. Then, in a chain of regular languages $\mathscr{L} = (\mathscr{L}_h \subseteq \cdots \subseteq \mathscr{L}_1)$, each $\mathscr{L}_i$ can be considered as an α-cut of a fuzzy regular language. Indeed, considering the lattice $L = (\{0, 1, \dots, h\}, \leq)$ (for $\leq$ being the usual order of the natural numbers), the language $\mathscr{L}_i = \{s \in \Sigma^* \mid \mathcal{L}(s) \geq i\}$ is the i-cut of the fuzzy language $\mathcal{L}_{\mathscr{L}} : \Sigma^* \to \{0, 1, \dots, h\}$ such that $\mathcal{L}_{\mathscr{L}}(s) = h$ for $s \in \mathscr{L}_h$, $\mathcal{L}_{\mathscr{L}}(s) = i$ for $1 \leq i < h$ iff $s \in \mathscr{L}_i \setminus \mathscr{L}_{i+1}$, and $\mathcal{L}_{\mathscr{L}}(s) = 0$ otherwise. Hence, each chain of regular languages $\mathscr{L} = (\mathscr{L}_h \subseteq \cdots \subseteq \mathscr{L}_1)$ corresponds to the fuzzy language $\mathcal{L}_{\mathscr{L}}$ so defined.

We recall that a *fuzzy automaton* over a finite alphabet Σ, in the complete lattice $\mathbf{L} = (L, \leq)$, is a tuple $A = (\mathsf{Q}, \{\mathsf{D}_a\}_{a\in\Sigma}, \mathsf{I}, \mathsf{F})$ where Q is the finite crisp set of states, each D_a is a fuzzy transition relation in $L^{\mathsf{Q}\times\mathsf{Q}}$, while $\mathsf{I} \in L^{\mathsf{Q}}$ and $\mathsf{F} \in L^{\mathsf{Q}}$ are the fuzzy sets of the initial and final states, respectively (see [14]).
The degree $(\mathcal{L}(A))(s)$ to which a fuzzy automaton accepts a word $s = a_1 \cdots a_u$ is defined as

$$(\mathcal{L}(A))(s) = \bigvee_{q_1,\cdots q_{n+1}\in Q} I(q_1) \wedge D_{a_1}(q_1, q_2) \wedge \cdots \wedge D_{a_u}(q_u, q_{u+1}) \wedge F(q_{u+1}).$$

The fuzzy set $\mathcal{L}(A)$ is the fuzzy language accepted by A.

Let us consider a Gödel automaton $\mathscr{A} = (Q, \delta, C, F)$ recognising a chain $\mathscr{L} = (\mathscr{L}_h \subseteq \cdots \subseteq \mathscr{L}_1)$. We define the fuzzy automaton $A = (\mathsf{Q}, \{\mathsf{D}_a\}_{a\in\Sigma}, \mathsf{I}, \mathsf{F})$, in the complete lattice $(\{0, 1, \dots h\}, \leq)$, as follows:

1. Q is the set of states in Q;
2. each D_a is such that $\mathsf{D}_a(p, q) = h$ iff $\delta(p, a) = q$, $\mathsf{D}_a(p, q) = 0$ otherwise;
3. $\mathsf{I}(q_i) = i$ iff $q_i \in C = (q_h > \cdots > q_1)$, $\mathsf{I}(q_i) = 0$ otherwise;
4. $\mathsf{F}(q) = h$ iff $q \in F$, $\mathsf{F}(q) = 0$ otherwise.

It is straightforward that the language accepted by A corresponds to $\mathcal{L}_{\mathscr{L}}$. Notice that in the definition of A the fuzzy sets of transitions and of final states are actually crisp sets.

In [26] the author introduces the notion of *deterministic* fuzzy automata, and proves that deterministic fuzzy automata and general non-deterministic fuzzy automata are equally powerful, in the sense that the two classes of automata recognise the same fuzzy languages. A fuzzy automaton A is *deterministic* iff the fuzzy sets of transitions and initial state are actually crisp, that is, $A = (\mathsf{Q}, \delta, s_0, \mathsf{F})$ where δ is a function $\delta : Q \times \Sigma \to Q$, and $s_0 \in Q$, while Q and F are as in the general definition of fuzzy automaton.

Let $A = (\mathsf{Q}, \delta, s_0, \mathsf{F})$ be a deterministic fuzzy automaton, recognising the fuzzy language $\mathcal{L}(A) \in \mathbf{L}^{\Sigma^*}$, with $\mathbf{L}$ being the lattice $(\{0, 1, \dots, h\}, \leq)$. We define the Gödel automaton $\mathscr{A} = (Q, \gamma, C, F)$ as follows:

1. $Q = \mathsf{Q} \times \mathbf{L}$ as a poset;
2. $\gamma((s,i),a) = (\delta(s,a),i)$ for each $s \in \mathsf{Q}$ and $i = 0,1,\dots,h$;
3. $C = \{s_0\} \times \mathbf{L}$;
4. $F = \{(s,i) \mid s \in \mathsf{Q},\ i = 0,1,\dots,h,\ \mathsf{F}(s) \geq i\}$.

A simple check shows that the chain of languages $\mathscr{L} = (\mathscr{L}_h \subseteq \cdots \subseteq \mathscr{L}_1)$ recognised by $\mathscr{A}$ is such that $\mathcal{L}_{\mathscr{L}} = \mathcal{L}(A)$.

We conclude that the chains of languages recognised by Gödel fortresses (that is, finite nested chains of regular languages) corresponds bijectively with the fuzzy languages (with values in finite chains) that are recognised by fuzzy automata. Whence we can add the purely logic-oriented Gödel fortresses (together with Gödel automata) to the set of equally powerful systems recognising fuzzy languages, such as fuzzy automata, deterministic fuzzy automata, nested systems of non-deterministic classical automata, nested systems of deterministic classical automata.

7. Conclusions

This study has extended to Gödel logic the logical descriptor for regular languages introduced in [13]. By leveraging the duality between finite forests and finite Gödel algebras, we established a correspondence between the new notions of Gödel automata, that recognise chains of regular languages, and Gödel fortresses that are defined by logical means.

The main approach used to characterise the languages recognised by Gödel fortresses, namely, the use of the dual equivalence of Gödel algebras with finite, combinatorially described objects (the finite forests), could be applied to other non-classical logics, in order to interrelate the logics themselves, their algebraic semantics and the insights yielded by the dual structures, towards the interpretation of non-classical fortresses as computational devices.

The authors thank Denisa Diaconescu and Tommaso Flaminio for many fruitful discussions.

CRediT authorship contribution statement

Stefano Aguzzoli: Writing – review & editing, Writing – original draft, Formal analysis; **Brunella Gerla:** Writing – review & editing, Writing – original draft, Formal analysis; **Sarah Nastasi:** Writing – review & editing, Writing – original draft, Formal analysis.

Data availability

No data was used for the research described in the article.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Supplementary material

Supplementary material associated with this article can be found in the online version at 10.1016/j.fss.2025.109749

References

[1] K. Gödel, Zum intuitionistischen Aussagenkalkül, Ergebnisse eines mathematischen Kolloquiums Heft IV (1933) 40.
[2] A. Horn, Logic with truth values in a linearly ordered Heyting algebra, J. Symbolic Logic 34 (1969) 395–408.
[3] P. Hájek, Metamathematics of Fuzzy Logic, 4 of *Trends in logic*, Kluwer Academic Publishers, 1998.
[4] F. Esteva, L. Godo, Monoidal t-norm based logic: towards a logic for left-continuous t-norms, Fuzzy Sets Syst. 124 (3) (2001) 271–288.
[5] P.T. Johnstone, Stone Spaces, 3 of *Cambridge Studies in Advanced Mathematics*, Cambridge University Press, 1982.
[6] W.J. Blok, D. Pigozzi, Algebraizable Logics, 396 of *Memoirs of the American Mathematical Society*, American Mathematical Society (AMS), 1989.
[7] O.M. D'Antona, V. Marra, Computing coproducts of finitely presented Gödel algebras, Ann. Pure Appl. Logic 142 (2006) 202–211.
[8] S. Aguzzoli, P. Codara, Recursive formulas to compute coproducts of finite Gödel algebras and related structures, in: 2016 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2016, Vancouver, Canada, 2016, p. 201–208.
[9] S. Aguzzoli, M. Mordonini, Counting homomorphisms between finite Gödel algebras, in: Proceedings of IPMU 2024, Lisbon, Portugal, Lecture Notes in Networks and Systems 1175, 2025, p. 365–377.
[10] S. Aguzzoli, B. Gerla, V. Marra, The automorphism group of finite Gödel algebras, in: Proceedings of the 40th International Symposium on Multiple-Valued Logic, Barcelona, Spain, 2010, p. 21–26.
[11] S. Aguzzoli, B. Gerla, Averaging the truth value of formulas in Gödel logic, in: 2023 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2023, Incheon, South Korea, 2023, pp. 1–6.
[12] S. Aguzzoli, B. Gerla, V. Marra, De Finetti's no-Dutch-book criterion for Gödel logic, Stud. Log. 90 (1) (2008) 25–41.
[13] S. Aguzzoli, D. Diaconescu, T. Flaminio, A logical descriptor for regular languages via stone duality, in: G. Ciobanu, D. Méry (Eds.), Theoretical Aspects of Computing – ICTAC 2014, Bucharest, Romania, Springer International Publishing, Cham, 2014, pp. 25–42.
[14] J.N. Mordeson, D.S. Malik, Fuzzy Automata and Languages: Theory and Applications, Computational mathematics series, Chapman and Hall/CRC, 2002.
[15] J.E. Hopcroft, R. Motwani, J.D. Ullman, Introduction to Automata Theory, Languages, and Computation, Addison-Wesley, 2000.
[16] S. Aguzzoli, S. Bova, B. Gerla, Free algebras and functional representation for fuzzy logics, in: P. Cintula, P. Hájek, C. Noguera (Eds.), Chapter IX of Handbook of Mathematical Fuzzy Logic, 38 of *Studies in Logic*, College Publications, London, 2011, pp. 713–791.
[17] S. Aguzzoli, B. Gerla, Normal forms and free algebras for some extensions of MTL, Fuzzy Sets Syst. 159 (2008) 1131–1152.
[18] B. Gerla, A note on functions associated with Gödel formulas, Soft Comput. 4 (2000) 206–209.

[19] S. Burris, H.P. Sankappanavar, A Course in Universal Algebra, Graduate Texts in Mathematics, Springer New York, NY, 2011.
[20] A. Horn, Free *L*-algebras, J. Symbolic Logic 34 (1969) 475–480.
[21] L.L. Esakia, Topological Kripke models, Soviet Math. Dokl. 15 (1974) 147–151.
[22] S. Aguzzoli, O.M. D'Antona, V. Marra, Computing minimal axiomatizations in Gödel propositional logic, J. Logic Comput. 21 (5) (2009) 791–812.
[23] J.R. Büchi, Weak second order arithmetic and finite automata, Zeitschrift für Mathematische Logik und Grundlagen der Mathematik 6 (1960) 66–92.
[24] C.C. Elgot, Decision problems of finite automata design and related arithmetics, Trans. Am. Math. Soc. 98 (1961) 21–52.
[25] B.A. Trakhtenbrot, Finite automata and monadic second order logic, Siber. Math. J. 3 (1962) 103–131. In Russian; English translation in Amer. Math. Soc. Transl. 59, 1966,23-55.
[26] R. Bělohlávek, Determinism and fuzzy automata, Inf. Sci. 143 (1–4) (2002) 205–209.