

On the Magnitude of Completeness Thresholds in Bounded Model Checking

Daniel Bundala
Department of Computer Science
Oxford University
Oxford, UK

Joël Ouaknine
Department of Computer Science
Oxford University
Oxford, UK

James Worrell
Department of Computer Science
Oxford University
Oxford, UK

Abstract—Bounded model checking (BMC) is a highly successful bug-finding method that examines paths of bounded length for violations of a given regular or ω -regular specification. A *completeness threshold* for a given model M and specification φ is a bound k such that, if no counterexample to φ of length k or less can be found in M , then M in fact satisfies φ . The quest for ‘small’ completeness thresholds in BMC goes back to the very inception of the technique, over a decade ago, and remains a topic of active research.

For a fixed specification, completeness thresholds are typically expressed in terms of key attributes of the models under consideration, such as their diameter (length of the longest shortest path) and especially their *recurrence diameter* (length of the longest loop-free path). A recent research paper identified a large class of LTL specifications having completeness thresholds *linear* in the models’ recurrence diameter [7]. However, the authors left open the question of whether linearity is in general even decidable.

In the present paper, we settle the problem in the affirmative, by showing that the linearity problem for both regular and ω -regular specifications (provided as automata and Büchi automata respectively) is PSPACE-complete. Moreover, we establish the following dichotomies: for regular specifications, completeness thresholds are either linear or exponential, whereas for ω -regular specifications, completeness thresholds are either linear or at least quadratic.

Index Terms—Bounded model checking, computer-aided verification, automata theory.

I. INTRODUCTION

Bounded model checking (BMC) was introduced in [1], [2] as a symbolic bug-finding method that searches for lasso-shaped counterexamples to an LTL formula in a given Kripke structure. Within three or four years following its introduction, it was found to have almost entirely replaced BDD-based model checking in the hardware industry, thanks largely to the huge advances made in SAT technology over the past 10 to 15 years.

The fundamental approach underpinning BMC is to look for counterexamples, or bugs, of bounded length. As such, an absence of counterexample is inconclusive; a genuine bug could still lurk deeper within the system. For this reason, from the very inception of the technique, researchers have attempted to turn BMC into a *complete* method with the ability also to guarantee the absence of counterexamples of any length. See, for instance, the original work of Biere *et al.* [2], or the 2008 Turing Award lecture of Ed Clarke [4], in which the problem is described as a topic of active research. See also the work

on cube enlargement techniques [8], circuit co-factoring [5], induction [10], and Craig interpolation [9].

In [2], Biere *et al.* observed that for simple safety properties of the form Gp , a *completeness threshold* is given by the *diameter* (longest distance between any two states) of the Kripke structure under consideration: indeed, if no counterexample to Gp of length at most the diameter of the system can be found, then no counterexample of any length can possibly exist. Likewise, for liveness properties such as Fq , the *recurrence diameter* (length of the longest loop-free path) of the Kripke structure can be seen to be an adequate completeness threshold. In a recent paper [7], Kroening *et al.* substantially extend these observations by identifying a large class of ω -regular specifications for which completeness thresholds *linear* in the recurrence diameter of the models can be effectively computed. This class consists of so-called *cliquey* Büchi automata, and subsumes among others the fragment of LTL consisting of unary next-free formulas. The authors also present examples of simple specifications having quadratic or even exponential completeness thresholds.¹

Unfortunately, [7] left as an open question whether the problem of determining if an ω -regular specification has a linear completeness threshold is decidable. In this paper, we answer the question affirmatively by showing that the linearity problem for both regular and ω -regular specifications (provided as automata and Büchi automata respectively) is PSPACE-complete; and when the completeness threshold is indeed linear, we provide effective bounds on the linearity constant. Moreover, we establish the following dichotomies: for regular specifications, completeness thresholds are either linear or exponential, whereas for ω -regular specifications, completeness thresholds are either linear or at least quadratic (and can be precisely quadratic or exponential).

In general, model checking can be applied to structures in which either states or edges are labelled. We have opted to present our work in the context of edge-labelled structures, also known as automata, in order to simplify our exposition; our main results also apply to state-labelled structures (Kripke structures), although the careful verification of this fact is a somewhat laborious exercise.

¹The precise definition of the magnitude of completeness thresholds is given in Sec. II.

II. NOTATION

We denote the empty word by ϵ . A *finite automaton* over alphabet Σ is a tuple (Q, I, Δ, F) where Q is the set of states, $I \subseteq Q$ is the set of initial states, $\Delta \in \mathcal{P}(Q \times \Sigma \times Q)$ is the transition relation and $F \subseteq Q$ is the set of final states. When we consider finite automata as acceptors of infinite words we refer to them as *Büchi automata*. If B is an automaton, then we use B to denote both the automaton and (assuming the context is clear) its set of states. In particular, we denote by $|B|$ the number of states of automaton B .

A *transition* is a tuple in the transition relation. A *path* π through an automaton B is a sequence of transitions $\pi = (e_i)_{i=1}^l$ where $l \in \mathbb{N} \cup \{\infty\}$ and the edges are consecutive. The length of π , denoted $|\pi|$, is equal to l , and the word spelled by π is denoted $\text{word}(\pi)$. A finite path π is *k-bounded* if $|\pi| \leq k$. An infinite path π is *lasso-shaped* and *k-bounded* if it can be written as $\pi = u.v^\omega$, where $|u| + |v| \leq k$. A path is *simple* if every state is visited at most once. A path starting in an initial state is *accepting* if it ends in a final state (for finite automata) or if some final state is visited infinitely often (for Büchi automata).

An *edge-labelled transition system* over Σ is a tuple (S, s_0, Δ) where S is the set of states, s_0 is the initial state and $\Delta \subseteq \mathcal{P}(Q \times \Sigma \times Q)$ is the transition relation. All models considered in this paper are edge-labelled transition systems and we refer to them simply as *models*.

Let M be a model. The *diameter* of M , denoted $d(M)$, is the length of a longest shortest path between any two reachable states of M . The *recurrence diameter* of M , denoted $\text{rd}(M)$, is the length of a longest loop-free (simple) path through M . For an automaton B having at least one accepting path, we define $\text{sap}(B) = \min\{k \mid B \text{ has a (lasso-shaped) } k\text{-bounded accepting path}\}$ to be the length of a *shortest accepting path*. Observe that $d(M) \leq \text{rd}(M) \leq |M|$ and, in case of finite automata, $\text{sap}(B) \leq d(B)$ where we extend the definition of diameter to automata in the natural way.

The *product* of a model $M = (Q, s_0, \Delta)$ with an automaton $B = (Q', I', \Delta', F)$, denoted $M \times B$, is an automaton over Σ defined to be $(Q'', I'', \Delta'', F'')$ where the two transition systems synchronise on edge transitions:

- $Q'' = Q \times Q'$
- $I'' = \{s_0\} \times I'$
- $\Delta'' = \{((q_1, q'_1), a, (q_2, q'_2)) \in Q'' \times \Sigma \times Q'' \mid (q_1, a, q_2) \in \Delta, (q'_1, a, q'_2) \in \Delta'\}$
- $F'' = Q \times F$.

Definition 1. An automaton B has a **linear completeness threshold** if there exists $c \in \mathbb{R}^+$ such that for all models M , if $M \times B$ has some accepting path, then $\text{sap}(M \times B) \leq c \cdot \text{rd}(M)$.

Definition 2. An automaton B has **at least quadratic/exponential completeness threshold** if there exists a sequence of models $(M_i)_{i=1}^{\infty}$ with $\text{rd}(M_i) \rightarrow \infty$ and a constant $c \in \mathbb{R}_+$ such that $\text{sap}(M_i \times B) \geq c \cdot \text{rd}(M_i)^2$ or $\text{sap}(M_i \times B) \geq 2^{c \cdot \text{rd}(M_i)}$, respectively.

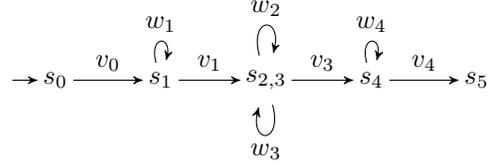


Fig. 1. A loop model.

In short, we say that B is linear, at least quadratic or at least exponential. Note that although we defined completeness thresholds using automata, it is a property purely of the *language* recognised by the automaton: B is linear iff for all models M the shortest word in $L(M) \cap L(B) \leq c \cdot \text{rd}(M)$.

If π is a path through M then $\pi(i)$ denotes the i -th vertex (starting from 1) of π . Then for $a < b \in \mathbb{N}$ the expression $\pi[a \dots b]$ denotes the subpath of π from index a to index b inclusively. The expression $\pi[a \dots)$ denotes the suffix of π starting at index a . The first and the last state of π are denoted $\text{first}(\pi)$ and $\text{last}(\pi)$ respectively. If τ is another path in M such that $\text{last}(\pi) = \text{first}(\tau)$ then $\pi\tau$ is the concatenation of π followed by τ . For states $(s_1, b_1), (s_2, b_2) \in M \times B$ let $\Pi_{(s_1, b_1)}^{(s_2, b_2)} = \{\pi \mid \pi \text{ is a path through } M \times B, \text{first}(\pi) = (s_1, b_1), \text{last}(\pi) = (s_2, b_2), \forall i < |\pi|. \pi(i) \neq (s_2, b_2)\}$ be the set of paths through M starting in (s_1, b_1) , ending in (s_2, b_2) and visiting the state s_2 only once. If ρ is a path in B such that $\text{word}(\rho) = \text{word}(\pi)$ then $\pi \otimes \rho$ denotes the unique path in $M \times B$ obtained by composing π and ρ componentwise.

III. REGULAR LANGUAGES

Throughout this section, we consider a fixed finite automaton B . Let M be a model. The structure of loops in M , as determined by the paths in the product $M \times B$, turns out to be crucial to linearity of B . For example, if π is a simple path in M , then it cannot possibly be longer than the recurrence diameter of M . In order to be longer than the recurrence diameter, π has to intersect itself and thus contain some loops. Roughly speaking, we will show that π needs approximately k reasonably well-behaved loops in order to be of length at least k times the recurrence diameter. These considerations motivate the introduction of models of the following special form, an example of which is shown in Fig. 1.

Definition 3. Given $n \in \mathbb{N}$, words $w_1, \dots, w_n \in \Sigma^+$ and words $v_0, v_1, \dots, v_n \in \Sigma^*$ we define a **loop model** as follows. The model contains (among others) states s_0, \dots, s_{n+1} . State s_0 is the initial state. For every $0 \leq i \leq n$ there is a path, called an **arc**, from s_i to s_{i+1} that spells the word v_i and for $1 \leq i \leq n$ a loop path spelling w_i is attached to s_i . If $v_i = \epsilon$ then we identify s_i with s_{i+1} .

Definition 4. Let $w \in \Sigma^*$. We define a transition relation R_w between states of B induced by w by $(b_1, b_2) \in R_w$ if $b_1 \xrightarrow{w} b_2$. And we define the set S_w of states of B such that $b \in S_w$ if $P(w) \cap L(b) \neq \emptyset$ where $P(w)$ is the set of all prefixes of w (including the empty string and w itself) and $L(b)$ is the set of words accepted by B starting in b .

The relations R_w and S_w satisfy the following properties.

Lemma 5. *Let $w, v \in \Sigma^*$ be words. Then $R_{wv} = R_v \circ R_w$ and $b \in S_{wv} \iff b \in S_w \vee \exists c \in S_v. (b, c) \in R_w$.*

The automaton B induces the following equivalence relation on Σ^* , cf. [3].

Definition 6. *Two words $w, v \in \Sigma^+$ are **equivalent**, written $w \sim v$, if the corresponding relations and sets are equal, $R_w = R_v$ and $S_w = S_v$. We say that a word w or loop w is **pumpable** if the equivalence class $[w]$ is infinite. We denote the index of \sim by C_B .*

From Lemma 5, we easily see that \sim is a congruence, i.e., $w \sim v$ and $x \sim y$ imply that $wx \sim vy$. Note also that $C_B \leq 2^{|B|^2 + |B|}$. If the automaton B is clear from the context, we drop B and write C instead of C_B .

Lemma 7. *Let $w \in \Sigma^*$. If w is not pumpable then $|w| \leq C$ and if w is pumpable then for any $K \in \mathbb{N}$ there is a word v such that $w \sim v$ and $K \leq |v| \leq K + C$.*

Proof: (Sketch) If w decomposes as $w = xyz$ such that $x \sim xy$ then $xy^kz \sim xyz$ for any $k \geq 0$. ■

Consider a loop model. We shall show below that only the number of pumpable loops taken by accepting paths is essential to nonlinearity of B . Also, some of the loops of a loop model may not be taken by every accepting path and are thus redundant. We further restrict only to models without redundant loops and by concatenating loops attached to the same state we can also assume that at most one loop is attached to every state.

Definition 8. *Let $M = (n, \vec{w}, \vec{v})$ be a loop model with n loops such that $|v_i| > 0$ for $i > 0$ and let $\rho = ((m_1, b_1) \dots (m_t, b_t))$ be a path in $M \times B$. Then we can associate the vector (x_1, \dots, x_n) with ρ where $x_i := |\{1 \leq j \leq t \mid m_j = s_i\}| - 1$ equals the number of times ρ takes the loop w_i (in the projection onto M). We say that ρ skips the loop w_i if $x_i \leq 0$. The model M is called an **irredundant loop model (for B)** if $|v_i| > 0$ for every $0 < i \leq n$, every loop w_i is pumpable, no accepting path skips a loop and $L(M) \cap L(B) \neq \emptyset$.*

The equivalence of two words $w \sim v$ has been specifically designed so that if a word is replaced by an equivalent one in an irredundant loop model, we obtain another irredundant loop model.

Lemma 9. *Let $M = (n, \vec{w}, \vec{v})$ be an irredundant loop model and let N be the model obtained by replacing w_i by x for some $x \sim w_i$ and $1 \leq i \leq n$. Then N is an irredundant loop model.*

Proof: (Sketch) Let π be an accepting path through $N \times B$ and suppose, to the contrary, that it skips some loop. We shall turn π into a path ρ through $M \times B$ that takes the loop w_i as many times as π takes x and is the same as π elsewhere.

If $\pi[k \dots l]$ for $k < l$ is one traversal of x by π then $\pi(k) = (s_i, b_k)$ and $\pi(l) = (s_i, b_l)$. Thus $b_k \xrightarrow{x} b_l$ in B . Since $w_i \sim x$, it holds that $b_k \xrightarrow{w_i} b_l$. So there is a path τ in B from b_k to

b_l . Let α be a path through M from s_i to s_i corresponding to one traversal of the loop w_i . Then replace $\pi[k \dots l]$ by $\alpha \otimes \tau$.

If π terminates inside the loop x then let $\pi(k)$ be the last occurrence of s_i in π . Then $\pi(k) = (s_i, b_k)$ and $\text{last}(\pi) = (m, b_l)$ for some m inside the loop x and b_l accepting state. Thus, $b_k \in S_x$ and since $w_i \sim x$, it holds that $b_k \in S_{w_i}$. So there is a path τ_2 in B from b_k to b_m for some accepting state b_m of B such that $\text{word}(\tau_2)$ is a prefix of w_i . Let α_2 be a path through M from s_i traversing the loop w_i and spelling $\text{word}(\tau_2)$. Then we replace $\pi[k \dots l]$ by $\alpha_2 \otimes \tau_2$ thereby obtaining a path ending in a final state. ■

Similarly, we also have:

Lemma 10. *Let $M = (n, \vec{w}, \vec{v})$ be an irredundant loop model and let N be the model obtained by replacing v_i by x for some $x \sim v_i$ and $0 \leq i \leq n$. Then N is an irredundant loop model.*

We now state the main theorem of this paper relating automata that have nonlinear completeness threshold and irredundant loop models. The third statement forms the crucial part of our decision procedure.

Theorem 11. *Let B be a finite automaton. Then the following are equivalent.*

- (a) B does not have a linear completeness threshold.
- (b) For every $k \in \mathbb{N}$ there exists an irredundant loop model with at least k loops.
- (c) There exists an irredundant loop model with L loops such that $2K \geq L > K$, where $K = 2^{2|B|}|B|$.

We first show that (b) \implies (a). Then we prove that (b) \iff (c) and justify the specific value of K . Finally, we establish that (a) \implies (b).

We begin by proving a stronger version of (b) \implies (a).

Theorem 12. *Let B be an automaton. If for every $k \in \mathbb{N}$ there exists an irredundant loop model with at least k loops then B 's completeness threshold is at least quadratic.*

Proof: Let $M = (n, \vec{w}, \vec{v})$ be an irredundant loop model with n loops. Then, using Lemma 7, change every w_i to y_i such that $w_i \sim y_i$ and $n \leq |y_i| \leq n + C$. Also, change every v_i to x_i such that $v_i \sim x_i$ and $|x_i| \leq C$. Denote the obtained model by M' . Lemmas 9 and 10 guarantee that M' is irredundant. Now, $\text{rd}(M') \leq 2(n+C) + (n+1)C \leq 2nC + 2nC + nC + nC = 6nC$ since a longest loop-free path visits at most two loops and traverses all v_i 's. On the other hand, $\text{sap}(M' \times B) \geq n^2$ as every accepting path traverses each of at least n loops and each loop is of length at least n . Thus,

$$\text{sap}(M' \times B) \geq n^2 \geq \frac{\text{rd}(M')^2}{36C^2}$$

Since n can be arbitrarily large, the result follows. ■

Next, we shall show that if there is an irredundant loop model with more loops than a certain critical threshold then there are irredundant loop models with arbitrarily many loops. Let $M = (n, \vec{w}, \vec{v})$ be an irredundant loop model. Recall that we denoted the state of M to which the loop w_i is attached

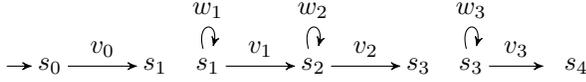


Fig. 2. A partition of an irredundant loop model.

by s_i and the initial state of B by b_0 . Then for every $0 < i < n$, the states of B can be assigned into the following two (overlapping) categories.

$$E_i := \{b \in B \mid \Pi_{(s_0, b_0)}^{(s_i, b)} \neq \emptyset\}$$

$$F_i := \{b \in B \mid \exists \pi \in \Pi_{(s_0, b_0)}^{(s_i, b)}. \pi \text{ skips some } w_j \text{ for } j < i\}$$

The set E_i collects the reachable states of B at the state s_i of M and the set F_i collects the states which are reachable by a path skipping a loop.

Furthermore, let π be an accepting path in $M \times B$. Then π visits every state $s_i \in M$ and so let $f(i)$ be the index of the first occurrence of s_i in π . If $n > 2^{2^{|B|}}|B|$ then, by the pigeonhole principle, there are indices $1 \leq i < j \leq n$ such that $E_i = E_j, F_i = F_j$ and there is a state $b \in B$ such that $\pi(f(i)) = (s_i, b)$ and $\pi(f(j)) = (s_j, b)$. Next, partition M into three loop models, corresponding to the prefix, pumpable segment and the suffix of M respectively: $X := (i-1, (w_1, \dots, w_{i-1}), (v_0, \dots, v_{i-1}))$, $Y := (j-i, (w_i, \dots, w_{j-1}), (\epsilon, v_i, \dots, v_{j-1}))$, $Z := (n-j+1, (w_j, \dots, w_n), (\epsilon, v_j, \dots, v_n))$. For example, the partition of an irredundant loop model with 3 loops split at $i=1, j=3$ is depicted in Fig. 2.

Thus, M can be written as $X \rightarrow Y \rightarrow Z$ where the last state of one part is identified with the first state of the next part. Similarly, let $\alpha = \pi[1 \dots f(i)]$, $\beta = \pi[f(i) \dots f(j)]$ and $\gamma = \pi[f(j) \dots]$. The following theorem shows that Y can be pumped while keeping the model irredundant.

Lemma 13. *The model $M' = X \rightarrow Y \rightarrow Y \rightarrow Z$ is irredundant and $\alpha\beta\beta\gamma$ is an accepting path through $M' \times B$.*

Proof: By contradiction, making case distinction on the place where an accepting path skips a loop.

Let i, j be as above. Observe that the states s_i, s_j of M' correspond to the first state of the first copy of Y and the first state of the second copy of Y , respectively. Let ρ be an accepting path in $M' \times B$ and suppose, to the contrary, that ρ does not take some loop w_k of M' . If ρ does not visit the state s_j then ρ is completely contained in $(X \rightarrow Y) \times B$ and so ρ is an accepting path in $M \times B$ skipping w_n . So suppose that ρ visits the state s_j and let $\rho(f(j)) = (s_j, t)$ be the state of ρ when it is first visited. Denote the suffix of ρ starting at $f(j)$ by $\tau_2 = \rho[f(j) \dots]$. Note that τ_2 is a path in $(Y \rightarrow Z) \times B$ that finishes in a final state. There are two possibilities: either $k < j$ and w_k is in $X \rightarrow Y$ or $k \geq j$ and w_k is in $Y \rightarrow Z$.

Suppose that $1 \leq k < j$. Then $t \in F_j$ since t is reachable by a path that skips w_k . By the choice of Y , $t \in F_i$ as well and so there is a path τ_1 from the initial state to (s_i, t) in $M \times B$ that skips a loop. Note that τ_1 is a path in $X \times B$ and

so $\tau_1\tau_2$ is an accepting path in $M \times B$ that skips some loop. But this is impossible as M is irredundant.

Now suppose that $j \leq k$. Then τ_2 is a path in $(Y \rightarrow Z) \times B$ that skips w_k . Also, t is reachable from the initial state and thus $t \in E_j$. By the choice of Y , $t \in E_i$ as well and so there is a path τ_1 from the initial state to (s_i, t) in $M \times B$. Note that τ_1 is a path in $X \times B$ and so $\tau_1\tau_2$ is an accepting path in $M \times B$ that does not take every loop. But this is impossible as M is irredundant.

Finally, note that α, β and γ are paths through $X \times B, Y \times B$ and $Z \times B$ respectively. By the construction $\text{first}(\beta) = (s_i, b)$ and $\text{last}(\beta) = (s_j, b)$. Thus β concatenated with itself gives a valid path through $(Y \rightarrow Y) \times B$. Since γ ends in a final state, $\alpha\beta\beta\gamma$ is an accepting path in $M' \times B$. ■

By modifying the above proof slightly, we can also show the following:

Lemma 14. *The model $M'' = X \rightarrow Z$ is irredundant and $\alpha\gamma$ is an accepting path through $M'' \times B$.*

Since we can pump as well as remove segments from large enough irredundant loop models, we have:

Theorem 15. *There are irredundant loop models with arbitrarily many loops if and only if there is an irredundant loop model $M = (n, \vec{w}, \vec{v})$ with $2^{2^{|B|}}|B| < n \leq 2^{2^{|B|+1}}|B|$.*

By flattening some of the repeated loops we shall prove in the following paragraphs that the nonlinearity of B is witnessed by very structured irredundant loop models. Observe that the values of E_{i+1} and F_{i+1} depend only on E_i, F_i, w_i and v_i . Therefore, $E_i = E_j$ and $F_i = F_j$ even for $M' = X \rightarrow Y \rightarrow Y \rightarrow Z$. Thus the same reasoning can be applied inductively to pump Y thereby obtaining a family of irredundant loop models of the form $X \rightarrow Y^* \rightarrow Z$.

Theorem 16. *For every $k \in \mathbb{N}$ the model $X \rightarrow Y^k \rightarrow Z$ is irredundant and $\alpha\beta^k\gamma$ is an accepting path in $(X \rightarrow Y^k \rightarrow Z) \times B$.*

Suppose that π traverses the loop w_i exactly x_i times. Let $v := v_i w_{i+1}^{x_{i+1}} v_{i+1} \dots w_{j-1}^{x_{j-1}} v_{j-1}$ be the unwinding of all but the first loop of Y according to β . That is, $\text{word}(\beta) = w_i^{x_i} v$. Finally, define Y' to be the loop model $(1, w_i^{x_i}, (\epsilon, v))$. Observe that Y' is only a submodel of Y and since $X \rightarrow Y^* \rightarrow Z$ is irredundant, so is $X \rightarrow (Y')^* \rightarrow Z$. In particular, $\text{word}(\alpha\beta^k\gamma) \in L((X \rightarrow Y'^k \rightarrow Z) \times B)$. Finally, by flattening the loops in X and Z we obtain a family of models of the form $x \rightarrow (Y')^* \rightarrow z$ where $x = \text{word}(\alpha)$ and $z = \text{word}(\gamma)$.

Theorem 17. *For every $k \in \mathbb{N}$, the model $N_k := x \rightarrow Y'^k \rightarrow z$ is irredundant.*

Denote the only remaining loop in Y' by w . The models N_k 's have very intricate structure. In particular, no accepting path through $N_k \times B$ takes two v 's in succession or takes x immediately followed by v . Consider the fractal-like models as shown in Fig. 3. The model is obtained by identifying the leaves of two complete binary trees, orienting the edges as

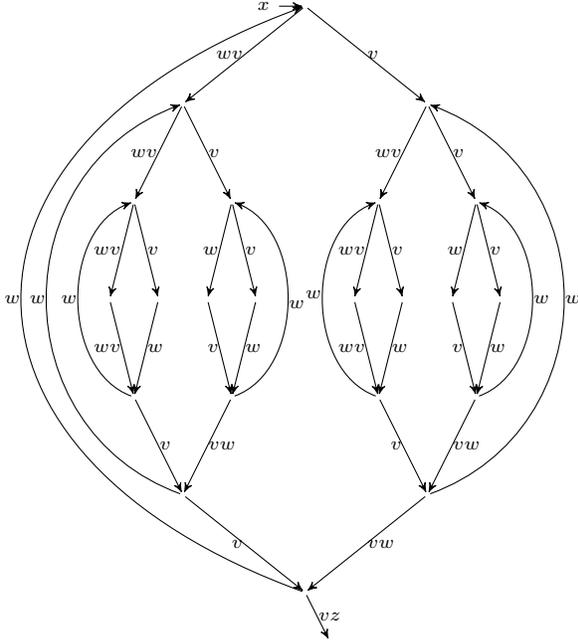


Fig. 3. Model with the shortest accepting path exponential in the recurrence diameter.

in the figure and by adding back edges from one tree to the other. Let M be the model in Fig. 3, ρ an accepting path in $M \times B$, and k the number of traversals of v by ρ . The path ρ can easily be modified into an accepting path in $N_k \times B$ spelling the same word. But using the properties of N_k stated above it follows that ρ recursively traverses the entire structure. The path ρ might stay in a submodel of M for a while, but eventually, it has to traverse every edge of M . Finally, observe that no loop-free path in M takes more than one edge going up. Therefore, the recurrence diameter is logarithmic in the number of edges. Thus, using Thm. 11, we have shown:

Theorem 18. *If the completeness threshold of an automaton B is not linear, then it must be at least exponential.*

This result is also optimal. Let M be a general (not necessarily irredundant) model and let V be the number of its states. Then $\text{sap}(M \times B) \leq |B|V$. Observe that by removing an edge from M the recurrence diameter of M never increases whereas $\text{sap}(M \times B)$ never decreases. Therefore, we can assume that M contains only edges on the shortest accepting path. Thus the out-degree and in-degree of every node in M is at most $|B|$. Hence, the number of states reachable in K steps or less from the initial state is at most $|B|^{K+1}$. Since all states are reachable, there must be a state at distance at least $\log_{|B|} V - 1$ from the initial state. Therefore, the diameter, and hence the recurrence diameter, of M is at least $\text{rd}(M) \geq \log_{|B|} V - 1 = \log_{|B|} (|B|V) - 2 \geq \log_{|B|} (\text{sap}(M \times B)) - 2$.

Theorem 19. *Let M be a model such that $L(M \times B) \neq \emptyset$. Then $\text{rd}(M) \geq \log_{|B|} (\text{sap}(M \times B)) - 2$.*

A. From General Models to Irredundant Loop Models

Finally, we prove the remaining implication (a) \implies (b) of Thm. 11. In this section, let M denote an edge-labelled transition system. The main idea of the proof is to take a projection onto M of an accepting path through $M \times B$ and, by identifying non-overlapping loops, folding it into a loop model. In order to control the folding we need the loops in the projection to be as simple as possible. This motivates the following definition:

Definition 20. *A path ρ through M is **locally minimal** if both the following conditions hold:*

- 1) *Every state s of M appears at most C times in ρ .*
- 2) *The distance between successive occurrences of the same state is at most C . In other words, for $i < j$, if $\rho(i) = \rho(j)$ and $\forall i < k < j. \rho(k) \neq \rho(i)$, then $j - i \leq C$.*

Definition 21. *A loop ρ of length n through M is **locally minimal** if both the following conditions hold:*

- 1) *If $\rho(i) = \rho(1)$ then $i = 1$ or $i = n$.*
- 2) *$\rho[2 \dots n - 1]$ is a locally minimal path.*

Although not loop-free, locally minimal paths and loops provide a bound on $\text{rd}(M)$.

Lemma 22. *Let ρ be a locally minimal path through M . Then $\text{rd}(M) \geq |\rho|/C^2$.*

Proof: Denote the length of ρ by n . We shall prove by induction on n that there is a loop-free path $\pi \subseteq \rho$ through M beginning with $\text{first}(\rho)$ of length at least n/C^2 .

The base case, $n = 1$, is trivial. For ρ longer than 1, let $\rho(k)$ be the last occurrence of $\rho(1)$ in ρ . Possibly, $k = 1$. Since ρ is locally minimal, it holds that $k \leq C^2$.

If $k = n$ then take $\pi = \rho(1)$. Then $|\pi| = 1$ and $n \leq C^2$. Otherwise, if $k < n$ then let π' be the path obtained by applying the induction hypothesis to $\rho[k + 1, \dots, n]$. Finally, take $\pi = \rho(1) \cdot \pi'$. Since π' begins with $\rho(k + 1)$ and $\rho(1) = \rho(k)$, the sequence π is a path. Moreover, since $\rho(1) \notin \rho[k + 1 \dots n]$, the path π is simple. Furthermore,

$$|\pi| = |\pi'| + 1 \geq \frac{n - k}{C^2} + 1 \geq \frac{n - C^2}{C^2} + 1 = \frac{n}{C^2} \quad \blacksquare$$

A similar lemma holds for locally minimal loops.

Lemma 23. *Let ρ be a locally minimal loop through M . Then $\text{rd}(M) \geq (|\rho| - 1)/C^2$.*

We now define the formal notion of a folding of a path through M .

Definition 24. *Given numbers $k_1, \dots, k_n > 0$, locally minimal paths $\alpha_0, \dots, \alpha_n$ and locally minimal loops $\beta_{1,1}, \dots, \beta_{1,k_1}, \dots, \beta_{n,1}, \dots, \beta_{n,k_n}$ satisfying for all i, j :*

- $\text{last}(\alpha_i) = \text{first}(\alpha_{i+1}) = \text{first}(\beta_{i+1,j}) = \text{last}(\beta_{i+1,j})$,
- $|\alpha_i| > 0$ for $0 < i < n$, and
- $|\beta_{i,j}| > C$,

a **normalisation** N with n loop bundles is a loop model given by arcs $\text{word}(\alpha_0), \dots, \text{word}(\alpha_n)$ with loops

$\text{word}(\beta_{i,1}), \dots, \text{word}(\beta_{i,k_i})$ in the i th loop bundle. A normalisation is **accepting** if $\text{first}(\alpha_0)$ is the initial state of M and $L(N \times B) \neq \emptyset$.

For example, we can think of the model in Fig. 1 as a normalisation with 3 loop bundles of size 1, 2, and 1. Observe that we can use locally minimal paths and loops to map every path through N to a corresponding path through M .

Lemma 25. *Let N be a normalisation. Then $L(N) \subseteq L(M)$.*

Then we use normalisations to bound $\text{sap}(M \times B)$ in terms of $\text{rd}(M)$.

Lemma 26. *Let N be an accepting normalisation with n loop bundles. Then $\text{sap}(M \times B) \leq 4n|B|C^2\text{rd}(M)$.*

Proof: The shortest accepting path traverses at most $|B|$ loops in every loop bundle. Thus, using Lemmas 22 and 23 to bound the lengths of α 's and β 's, we obtain

$$\begin{aligned} \text{sap}(M \times B) &\leq \text{sap}(N \times B) \\ &\leq (n+1)C^2\text{rd}(M) + n|B|(C^2\text{rd}(M) + 1) \\ &\leq 4n|B|C^2\text{rd}(M). \end{aligned}$$

In particular, if there exists $n \in \mathbb{N}$ such that for every model M there is an accepting normalisation with fewer than n loop bundles then B is linear. We therefore have:

Theorem 27. *If B does not have a linear completeness threshold, then for every $k \in \mathbb{N}$ there is a model M_k such that $L(M_k \times B) \neq \emptyset$ and every accepting normalisation of M_k has at least k loop bundles.*

If $L(M \times B) \neq \emptyset$ then by instantiating the following result from the initial state to a reachable final state we obtain an accepting normalisation.

Theorem 28. *Let states $m_1, m_2 \in M$ and $w \in \Sigma^*$ be such that $m_1 \xrightarrow{w} m_2$. There exists a normalisation N such that $|N| \leq |w|$, $\text{first}(\alpha_0) = m_1$, $\text{last}(\alpha_n) = m_2$ and there is a path π from the initial state of N to the last state of N such that $\text{word}(\pi) \sim w$, where the α_i 's refer to the locally minimal paths from Def. 24.*

Proof: Let ρ be the shortest path in M from m_1 to m_2 such that $\text{word}(\rho) \sim w$. Clearly, $|\rho| \leq |w|$ and every state of M appears at most C times in ρ .

We prove by induction on the length of ρ that there is a normalisation N such that $|N| \leq |\rho|$, $\text{first}(\alpha_0) = \text{first}(\rho)$, $\text{last}(\alpha_n) = \text{last}(\rho)$ and there is a path π from the initial state of N to the last state of N such that $\text{word}(\rho) = \text{word}(\pi)$.

If ρ satisfies 2 from Def. 20 of locally minimal paths then we are done. Simply take $n = 0$ and $\alpha_0 = \rho$. Otherwise, pick i, j with minimal j such that $\rho(i) = \rho(j)$, $i < j - C$ and $\forall i < k < j. \rho(k) \neq \rho(i)$. Then $\rho[i \dots j]$ is a locally minimal loop.

If $i = 1$ then let N' be the normalisation obtained by applying the inductive hypothesis to $\rho[j \dots]$. And add $\rho[1 \dots j]$ into the first loop bundle of N' thereby obtaining N .

If $i > 1$ then note that $\rho[1 \dots i]$ is a locally minimal path. Let N' be the normalisation obtained by applying the inductive hypothesis to $\rho[i \dots]$. And set $N = \rho[1 \dots i] \rightarrow N'$. ■

Corollary 1. *Let M be a model such that $L(M \times B) \neq \emptyset$. Then an accepting normalisation exists.*

Finally, we show that the smallest accepting normalisation of M_k gives rise to an irredundant loop model with at least k pumpable loops.

Theorem 29. *Let B be nonlinear. Then for every $k \in \mathbb{N}$ there exists an irredundant loop model with at least $k - 1$ loops.*

Proof: Let M_k be as defined above and let N be the smallest accepting normalisation of M_k . By the choice of M , N has at least k loop bundles. Recall that k_i 's denote the number of loops in the i -th loop bundle. First, we show that every accepting path in $N \times B$ takes every loop at least once.

Suppose, to the contrary, that there is an accepting path π in $N \times B$ that skips some loop $w_{i,j}$. Let N' be obtained from N by removing the loop $w_{i,j}$ and the corresponding path $\beta_{i,j}$.

If $k_i > 1$ then N' is an accepting normalisation smaller than N . But this is impossible. If $k_i = 1$ then we have eliminated the i -th loop bundle from N thereby concatenating v_{i-1} with v_i and α_{i-1} with α_i . The concatenated path might not be locally minimal in M ; however, we can replace it by a smaller normalisation thanks to Thm. 28. The net result is a compound accepting normalisation that is smaller than N , again yielding a contradiction. Thus every accepting path takes every loop.

Second, we show that it is possible to transform N so that every loop bundle is of size 1. Fix an accepting path π through $N \times B$ and recall that $w_{i_1}, \dots, w_{i_{k_i}}$ are the loops in the i th loop bundle of N . Let w be the word spelled by π while traversing through $w_{i_1}, \dots, w_{i_{k_i}}$ in $N \times B$. Then we remove $w_{i_1}, \dots, w_{i_{k_i}}$ from N and replace them by the single loop spelling w . By Def. 24 of a normalisation, each w_i is longer than C . Hence, $|w| > C$. By applying this transformation to every loop bundle, we obtain a new model N' . If $|\alpha_n| = 0$ then $v_n = \epsilon$. So let w_n be the last loop of N' . We remove w_n from N' and append the word w_n to v_{n-1} so that the last arc is non-null.

Note that $\text{word}(\pi) \in L(N')$. Since N' is just a submodel of N , every accepting path through $N' \times B$ traverses every loop. And so N' is an irredundant loop model. ■

B. Decision Procedure

In this section, we present a PSPACE decision procedure to determine whether a given automaton has a linear completeness threshold. If the automaton is linear we further show how to bound the linearity constant c from Def. 1. The section finishes by giving a corresponding proof of PSPACE-hardness.

Let $K = 2^{2|B|}|B|$. The decision procedure works by guessing a large enough loop model and then checking, on-the-fly, that the model is irredundant. Thm. 11 guarantees that it suffices to search for a loop model with number of loops between K and $2K$.

The algorithm first nondeterministically chooses the number of loops n in the model and then it keeps updating the set E_i of states reachable by a path in $M \times B$ and F_i of states reachable by a path skipping a loop. See the text before Lemma 13 for the precise definition. To update these sets, the algorithm guesses a loop $w \in \Sigma^*$ and an arc $v \in \Sigma^*$. Instead of storing w and v , it calculates the relations R_w, R_v and sets S_w, S_v . These sets and relations are calculated by guessing the words letter by letter, and using Lemma 5 to update the sets incrementally.

The algorithm needs to ensure that w is pumpable. However, Lemma 7 guarantees that it suffices to nondeterministically select a word of length between C and $2C$.

Note that only polynomially many bits in $|B|$ are needed to store R_w and S_w and since C and K are singly exponential in $|B|$, only polynomially many bits are needed to store n or the length of w . Also, only $|B|$ bits are needed to store E and F . Thus, the algorithm can be implemented in nondeterministic polynomial space. Finally, the algorithm ensures that a final state of B is visited only at the end.

Algorithm 1 Decision procedure for finite automata

```

 $n \leftarrow$  guess a number  $\in (K, 2K]$ 
 $R_{v_0}, S_{v_0} \leftarrow$  guess a word
 $E \leftarrow R_{v_0}(\{s_0\}$  - the initial state of  $B$ )
 $F \leftarrow \emptyset$ .
for  $i = 1$  to  $n$  do
   $R_w, S_w \leftarrow$  guess a pumpable word
   $R_v, S_v \leftarrow$  guess a word
   $E' \leftarrow \bigcup_{0 \leq k \leq |B|} R_{w^k v}(E)$ 
   $F' \leftarrow \bigcup_{0 \leq k \leq |B|} R_{w^k v}(F) \cup R_v(E)$ 
  if  $S_{w^k} \cap E \neq \emptyset \vee (i < n \wedge \bigcup_{0 \leq k \leq |B|} S_{w^k v} \cap E \neq \emptyset) \vee (i = n \wedge S_v \cap F \neq \emptyset)$  then
    return false {A final state is reachable before the end}
  end if
   $(E, F) \leftarrow (E', F')$ 
end for
return  $(E \setminus F) \cap \text{Final} \neq \emptyset$ 

```

Suppose that we determine that B has a linear completeness threshold. Then, by Thm. 15, we know that every irredundant loop model has at most K loops and so by instantiating Algorithm 1 for $n = 0 \dots K$ we can calculate the maximum number L of loops in an irredundant loop model. We claim that every model M has an accepting normalisation with at most $L + 1$ loop bundles. For otherwise, there would be a model M_{L+2} such that every accepting normalisation of M_{L+2} has at least $L + 2$ loop bundles. But then by the argument in the proof of Thm. 29 there would be an irredundant loop model with $L + 1$ loops, which is impossible. Thus, every model M has an accepting normalisation with at most $L + 1$ loop bundles. Applying Lemma 26, we get that $\text{sap}(M \times B) \leq 4(L + 1)|B|C^2 \text{rd}(M)$.

Observe that K depends only on $|B|$ and so even without calculating L it is possible to bound the linearity constant c by $c \leq 4(K + 1)|B|C^2 \text{rd}(M)$. Thus the problem of finding a

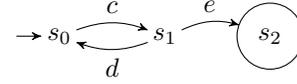


Fig. 4. Automaton with exponential completeness threshold

bound for c is also PSPACE-hard as we show next.

Theorem 30. *It is PSPACE-hard to determine whether an automaton B has a linear completeness threshold.*

Proof: The proof is by reduction from the universality problem for nondeterministic automata, well-known to be PSPACE-hard [6]. Let A be an automaton over alphabet Σ . We transform A into an automaton B over alphabet $\Sigma \cup \{\#\}$, where $\# \notin \Sigma$, such that A is universal if and only if B has a linear completeness threshold.

Let C be any automaton with non-linear completeness threshold (e.g., Fig. 4) and denote the models witnessing nonlinearity by $(M_i)_{i \in \mathbb{N}}$. Let D be a single-state automaton accepting Σ^* . Define automaton B to be the disjoint union of A , C and D , with additional $\#$ -labelled transitions from each non-accepting state of A to the initial state of C and from each accepting state of A to the initial state of D . Then, make all states of A in B non-accepting. B can clearly be constructed in polynomial time.

Now if A is universal then $L(B) = \Sigma^* \# \Sigma^*$. We claim that B has a linear completeness threshold in this case. Let M be a model such that $L(M \times B) \neq \emptyset$. Let $m \xrightarrow{\#} m'$ be a reachable $\#$ transition in M and let π be the shortest path from the initial state of M to m . By taking a prefix of π if necessary, we can assume that $\# \notin \text{word}(\pi)$. Then $|\pi| \leq \text{diam}(M) \leq \text{rd}(M)$ and $\text{word}(\pi) \# \in L(M) \cap L(B)$.

On the other hand, suppose that A does not accept some word $w \in \Sigma^*$. We claim that the models $(w \# \rightarrow M_i)_{i \in \mathbb{N}}$, obtained by attaching the last state of a path spelling $w \#$ to the initial state of M_i , witness nonlinearity of B .

Let π be an accepting path in $(w \# \rightarrow M_i) \times B$ and write $\pi = \pi_1 \# \pi_2$. Since A rejects w , the $\#$ -transition in π must go from a rejecting state of A to the initial state of C . Thus π_2 is an accepting path in $M_i \times C$. Therefore $|\pi_2| \geq \text{sap}(M_i \times C)$ and the claim follows since w is of a constant fixed size. ■

IV. ω -REGULAR LANGUAGES

We now sketch how to extend the results from regular languages to ω -regular languages. In this section, let B be a Büchi automaton. Our aim is to bound the length of the shortest lasso-shaped path in terms of the recurrence diameter. The equivalent of an irredundant loop model that is suitable for ω -regular languages is depicted in Fig. 5. The noose consists of one big loop with several (3 in Fig. 5) nested loops. We say that a model M of this form is an irredundant ω -loop model if for every accepting lasso shaped path $\pi_1 \pi_2^\omega$ it holds that π_1 traverses every loop of the stem and π_2 traverses every loop of the noose. Note that π_2 may traverse the noose more than once and in general, a single traversal of the noose might not

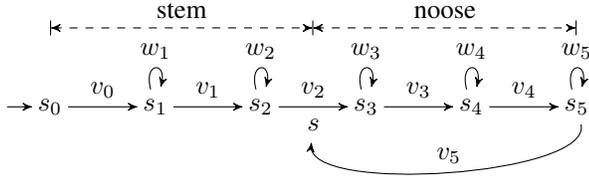


Fig. 5. General form of a non linear model for ω -regular languages.

traverse all loops inside the noose. Also note that the shortest lasso-shaped accepting path takes at most $|B|$ traversal through the noose.

In case of regular languages, a path is accepting if it ends in a final state. However, for ω -regular languages an accepting path needs to visit a final state infinitely often. Therefore, we have to modify the equivalence of two words to reflect this so that, if we replace a word by an equivalent one in an irredundant ω -loop model, we obtain a new irredundant ω -loop model.

Definition 31. Given $w \in \Sigma^*$ we define a binary relation T_w on B by $(b_1, b_2) \in T_w$ if there is a path ρ through B that begins in b_1 , finishes in b_2 , visits a final state and spells w .

Definition 32. Given $w, v \in \Sigma^*$, the words are equivalent, written as $w \sim v$, if $R_w = R_v$ and $T_w = T_v$.

Thm. 11 on the relationship between finite automata with nonlinear completeness threshold and irredundant loop models has a direct counterpart in terms of Büchi automata and irredundant ω -models.

Theorem 33. Let B be a Büchi automaton. Then the following are equivalent.

- (a) B does not have a linear completeness threshold.
- (b) For every $k \in \mathbb{N}$ there exists an irredundant ω -loop model with at least k loops.
- (c) There exists an irredundant ω -loop model with L loops such that $2K_\omega \geq L > K_\omega$, where $K_\omega = 2^{4|B|^2}$.

Similarly to regular languages, we prove that condition (b) is equivalent to condition (a) and condition (c). By pumping pumpable loops we can show the equivalent of Thm. 12 that irredundant ω -models of arbitrary size witness that B has at least quadratic completeness threshold.

Theorem 34. If for every $k \in \mathbb{N}$ there exists an irredundant ω -loop model with at least k loops then B has at least quadratic completeness threshold.

Recall that to show the analogous statement of (b) \iff (c) we carefully split the given irredundant loop model into three pieces $X \rightarrow Y \rightarrow Z$ so that the new model $X \rightarrow Y \rightarrow Y \rightarrow Z$ is still irredundant. We proceed similarly for irredundant ω -models. Suppose that M is an irredundant ω -loop model and M has more than K_ω loops. Then there are two possibilities. Either most of the loops are in the stem of M or in the noose. In the former case, we use the techniques developed for finite

languages to increase the number of loops. In the latter case, we extend the methods to pump the noose. Recall that for every s_i , the set E_i denoted the states of B reachable at s_i and the set $F_i \subseteq E_i$ consisted of states reachable by a path skipping a loop. Denote the state where the stem and the noose meet by s . Then a lasso-shaped path may traverse through the noose more than once and so we consider similar sets starting from (s, b) for every state $b \in B$. By symmetry, we will also need information about states of (s_i, b) from which it is possible to reach (s, b) for every $b \in B$. The following four sets are sufficient:

$$\begin{aligned}
 E_i^p &:= \{b \in B \mid \Pi_{(s,p)}^{(s_i,b)} \neq \emptyset\} \\
 F_i^p &:= \{b \in B \mid \exists \pi \in \Pi_{(s,p)}^{(s_i,b)} . \pi \text{ visits a final state}\} \\
 G_i^q &:= \{b \in B \mid \Pi_{(s_i,b)}^{(s,q)} \neq \emptyset\} \\
 H_i^q &:= \{b \in B \mid \exists \pi \in \Pi_{(s_i,b)}^{(s,q)} . \pi \text{ visits a final state}\}
 \end{aligned}$$

Analogously to the finite case, we have a theorem to the effect that if there are $i < j$ such that $(E_i^p, F_i^p, G_i^q, H_i^q) = (E_j^p, F_j^p, G_j^q, H_j^q)$ for every $p, q \in B$ then the segment between i and j can be pumped. By the pigeonhole principle, this is guaranteed to happen if the number of loops in the noose is greater than $2^{4|B|^2}$.

Theorem 35. For every $k \in \mathbb{N}$ there is an irredundant ω -loop model with at least k loops if and only if there is an irredundant ω -loop model with n loops where $2K_\omega \geq n > K_\omega$.

At this stage in the theory of regular languages, we were able show that if an automaton is not linear then it has at least exponential completeness threshold. This construction, however, does not work for infinite strings. In the finite case, it is always true that a final state of B is visited only at the very end. However, in the case of infinite words, we may not rule out the possibility that a final state is visited during the traversals of w 's and v 's of which there are only finitely many. And so it is possible that there is an accepting path which loops forever in a submodel of the model from Fig. 3. In fact, we show in Thm. 38 that there is a simple Büchi automaton with precisely quadratic completeness threshold. On the other hand, there are still Büchi automata with exponential completeness threshold, similar to that depicted in Fig. 4.

Given a general model M , recall that we used normalisations (Def. 24) to show that nonlinearity of an automaton is always witnessed by irredundant loop models in case of finite languages. A normalisation N is a loop model that embeds into M and the embedding is supplied by a path in M so that loops and arcs of N correspond to locally minimal paths in M . A similar argument works for ω -regular languages. We define an analogous notion of ω -normalisation P , which is a ω -loop model that embeds into M . The embedding again arises from a path in M and ensures that loops and arcs of P map into locally minimal paths in M . If $\pi = \pi_1 \pi_2^\omega$ is a lasso-shaped path then the embedding of the stem and noose arise from π_1 and π_2 respectively. Then, we use ω -normalisation to study M .

As in Lemma 26, for an accepting ω -normalisation with k loop bundles it holds that $\text{sap}(M \times B) \leq \text{sap}(P \times B) \leq f(k)\text{rd}(M)$ is bounded by a function of the number of loop bundles in P . Therefore, as in the Thm. 27, if a Büchi automaton is nonlinear then for every $k \in \mathbb{N}$ there must exist a model such that every accepting ω -normalisation has at least k loops. Then, analogously to the Thm. 29, the ω -normalisation with the smallest noose, and smallest stem in case of a tie, gives rise to an irredundant ω -loop model.

Theorem 36. *Let B be a Büchi automaton with nonlinear completeness threshold. Then for every $k \in \mathbb{N}$ there exists an irredundant ω -loop model with at least k loops.*

A. PSPACE Decision Procedure for ω -Regular Languages

Recall that in order to obtain irredundant ω -models with arbitrarily many loops we took a model with at least K_ω loops and depending on the concentration of loops, we either pumped the stem or the noose. We make this distinction in the decision procedure, which begins by nondeterministically guessing which of the two cases holds.

In the case of large stem, we can assume, by unwinding the noose if necessary, that the noose is a simple loop without any nested loops. Then we can reuse with minor modification the algorithm for regular languages to guess large enough stem and hence irredundant ω -loop model.

In the case the noose is being pumped, we can assume, by unwinding the stem if necessary, that the stem does not contain any loops. And so the algorithm only needs to check the existence of a large noose. The algorithm makes use of the following fact. If a noose has more than $|B|K_\omega$ loops and every accepting path traverses every loop then, since we can always restrict to paths traversing the noose at most $|B|$ times, at least K_ω loops are traversed during some single traversal of the noose.

The algorithm guesses the noose containing between K_ω and $2|B|K_\omega$ loops loop-by-loop. Denote the state where the stem and the noose meet by s . Then for each pair of states $p, q \in B$ the algorithm iteratively calculates two numbers A_{pq} and B_{pq} denoting the minimum number of loops on some single traversal of the noose from state (s, p) to state (s, q) with and without visiting a final state, respectively. Then, it uses these values to check whether for some reachable t there is a path from (s, t) to (s, t) traversing the noose possibly several times and visiting a final state so that during every single traversal of the noose it visits less than K_ω loops. This can be done by iterating over all paths traversing the noose at most $|B|$ times. If there is no such path then every accepting path through the noose traverses at least K_ω loops. So we can pick a subset of cardinality at least K_ω of loops so that all loops in S are traversed by every accepting path. Provided there is some accepting path, we have a witness for a pumpable noose.

The decision procedure needs to make one more check. It must ensure that it is impossible to remain stuck inside the noose forever and accept. This can also be done on-the-fly.

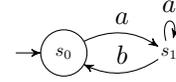


Fig. 6. A quadratic Büchi automaton.

Since K_ω is singly exponential in $|B|$ and we can assume that the noose and every loop is taken at most $|B|$ times, only polynomially many in $|B|$ bits are needed to store A_{pq}, B_{pq} and all intermediate values.

Finally, observe that by taking C to be a nonlinear Büchi automaton in (Thm. 30) we obtain the corresponding hardness result.

Theorem 37. *It is PSPACE-hard to determine whether a Büchi automaton B has a linear completeness threshold.*

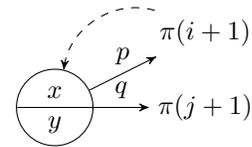
Unlike the case of regular languages, there are automata with completeness threshold strictly between linear and exponential.

Theorem 38. *The automaton in Fig. 6 is exactly quadratic.*

Proof: A family of irredundant ω -models witnessing at least quadratic completeness threshold has $w = a^*$ and $v = b$.

Let M be a model such that $L(M \times B) \neq \emptyset$ and let $\alpha\beta^\omega$ be the shortest accepting lasso-shaped path in $M \times B$. By proving results on the structure of loops of M traversed by α and β we show that the model induced by π is almost an irredundant ω -loop model.

Suppose that there are $i < j$ such that $\alpha(i) = (m, x)$ and $\alpha(j) = (m, y)$. Let p be the label of the edge from $\alpha(i)$ to $\alpha(i+1)$ and q be the label of the edge from $\alpha(j)$ to $\alpha(j+1)$ —if $\alpha(j)$ is the last state of α then take $\beta(1)$ instead. Since s_0 cannot be followed by b there are 9 possibilities for the tuple (x, y, p, q) .



If $x = y$ then $\alpha[1 \dots i]\alpha[j \dots]\beta^\omega$ is a shorter lasso-shaped path. Thus, no such loop appears in α .

If $q = a$ then $\alpha(j+1) = (n, s_1)$ for some $n \in M$ such that m is connected to n by an edge. But this state is reachable from $\alpha(i)$ and so the path $\alpha[1 \dots i]\alpha[j+1 \dots]\beta^\omega$ is a shorter accepting lasso shaped path. Thus, no such loop appears in α .

Finally, we are left with the situation where all loops satisfy $x = s_0, y = s_1, p = a$ and $q = b$. Now, $\alpha(i+1) = (n, s_1)$ which is a possible successor of $\alpha(j)$. So if there is some b between $\alpha(i)$ and $\alpha(j)$ then $\alpha[1 \dots i]\alpha[i+1 \dots j]^\omega$ is a shorter lasso-shaped accepting path. Hence, there cannot be any b between $\alpha(i)$ and $\alpha(j)$. That is, the word spelled by $\alpha[i \dots j]$ is a^{j-i} and since α is minimal, the projection of $\alpha[i \dots j]$ onto M is a simple loop. Therefore, there are no

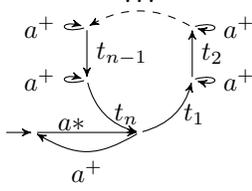


Fig. 7. Possible shape of the noose of the model induced by the path π from Thm. 38. Each t_i begins and ends with b , $t_i \in b + b\Sigma^*b$

nested loops in the projection of α onto M and so the model induced by α is a loop model.

This concludes α . Now suppose that there are $i < j$ such that $\beta(i) = (m, x)$ and $\beta(j) = (m, y)$ and it is not the case that $i = 1$ and $j = |\beta|$. Let p be the label of the edge from $\beta(i)$ to $\beta(i + 1)$ and q be the label of the edge from $\beta(j)$ to $\beta(j + 1)$ —if $\beta(j)$ is the last state of β then we take $\beta(1)$ instead. As above, there are 9 possibilities for (x, y, p, q) .

If $(x = y)$ or $(x = s_0, y = s_1, p = q = a)$ or $(x = s_1, y = s_0, p = q = a)$. Then it is possible to show that either by staying in the loop $\beta[i \dots j]$ forever or by eliminating it from β we obtain a shorter accepting path. So loops of this type do not appear in β .

If $x = s_0, y = s_1, p = a$ and $q = b$ then as above, it is possible to show that $\text{word}(\beta[i \dots j]) = a^{j-i}$, the projection of $\beta[i \dots j]$ onto M is a simple loop. So if there are only loops of this form in β then the model induced by β is a loop model with the first and the last state identified—a simple noose.

Finally, suppose that there is a loop of the form $x = s_1, y = s_0, p = b$ and $q = a$. Observe that $\beta(j + 1) = (n, s_1)$ for some $n \in M$ and so $\beta(j + 1)$ is a valid successor of $\beta(i)$. If there is a b in $\text{word}(\beta[j \dots j+1] \dots \beta[1 \dots i])$ then we can skip the loop thereby obtaining a shorter lasso-shaped accepting path $\alpha(\beta[1 \dots i]\beta[j+1 \dots j])^\omega$. So there is no b on that path, hence the word it spells consists only of a 's and so the projection of the path $\beta[j \dots j+1] \dots \beta[1 \dots i]$ onto M is a simple loop. Suppose that there is another loop of the type $x = s_1, y = s_0, p = b, q = a$ at position $k \dots l$ where $i \leq k < l \leq j$ such that $(i, j) \neq (k, l)$. Now, we know that $\beta(i + 1)$ and $\beta(j)$ are accepting. Since the loop $\beta[k \dots l]$ is properly nested in the loop $\beta[i \dots j]$ the removal of the loop $\beta[k \dots l]$ leaves at least one of these two states which gives rise to a shorter lasso-shaped accepting path. Therefore, there are no nested loops inside $\beta[i \dots j]$ of this form. Thus, all nested loops are as in the previous paragraph ($x = s_0, y = s_1, p = a, q = b$). It follows that the submodel of M induced by $\beta[i \dots j]$ is a loop model and the projection of β^ω onto M is as in Fig. 7.

So, α and β correspond to traversals of (almost) loop models and the models of this shape are clearly at most quadratic. ■

V. CONCLUDING REMARKS

This paper settles the main open questions listed in Sec. 6 of [7]: it is decidable, and in fact PSPACE-complete, whether a regular (resp. ω -regular) specification has a linear completeness threshold, provided the specification is given as an au-

tomaton (resp. Büchi automaton). Moreover, two dichotomies are at play: a regular specification either has a linear or an exponential completeness threshold, whereas an ω -regular specification has a completeness threshold that is either linear or at least quadratic.

As mentioned in the Introduction, these results also apply to state-labelled automata and models, by adapting our approach appropriately.

Several questions however remain. We conjecture that in the case of ω -regular specifications, there is in fact a *trichotomy*: completeness thresholds are either linear, precisely quadratic, or precisely exponential. We believe that such a result could be obtained by pushing further the techniques which we have developed in this paper.

Another interesting question is the complexity of determining whether an ω -regular specification has a linear completeness threshold, *assuming the specification is provided as an LTL formula*. One immediately obtains an EXPSPACE upper bound through the translation of LTL formulas into (at most) exponentially-sized Büchi automata. We were only able to establish a PSPACE lower bound for LTL and we conjecture that PSPACE in fact suffices. Note also that the PSPACE-hardness result relies on the automaton under consideration being nondeterministic. We leave open the question of complexity of the decision procedure for *deterministic* automata.

If an automaton has linear completeness threshold then it would be desirable to be able to estimate the linearity constant as precisely as possible. Although we provided upper bounds, these appear fairly loose from both practical and theoretical standpoints. The problem of *calculating and/or approximating* the constant more closely remains open.

Acknowledgements. This research was supported financially by EPSRC.

REFERENCES

- [1] A. Biere, A. Cimatti, E. Clarke, O. Strichman, and Y. Zhu, “Bounded model checking,” *Advances in Computers*, vol. 58, 2003.
- [2] A. Biere, A. Cimatti, E. Clarke, and Y. Zhu, “Symbolic model checking without BDDs,” in *Proceedings of TACAS*, vol. 1579. Springer LNCS, 1999.
- [3] J. R. Büchi, “On a decision method in restricted second order arithmetic,” in *Proceedings of the 1960 International Congress on Logic, Methodology and Philosophy of Science*, E. Nagel, P. Suppes, and A. Tarski, Eds., 1962.
- [4] E. M. Clarke, E. A. Emerson, and J. Sifakis, “Model checking: Algorithmic verification and debugging,” *CACM*, vol. 52, 2008.
- [5] M. Ganai, A. Gupta, and P. Ashar, “Efficient SAT-based unbounded symbolic model checking using circuit cofactoring,” in *Proceedings of ICCAD*. IEEE Computer Society, 2004.
- [6] J. E. Hopcroft and J. D. Ullman, *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley, 1979.
- [7] D. Kroening, J. Ouaknine, O. Strichman, T. Wahl, and J. Worrell, “Linear completeness thresholds for bounded model checking,” in *Proceedings of CAV*, vol. 6806. Springer LNCS, 2011.
- [8] K. McMillan, “Applying SAT methods in unbounded symbolic model checking,” in *Proceedings of CAV*, vol. 2404. Springer LNCS, 2002.
- [9] —, “Interpolation and SAT-based model checking,” in *Proceedings of CAV*, vol. 2725. Springer LNCS, 2003.
- [10] M. Sheeran, S. Singh, and G. Stålmarck, “Checking safety properties using induction and a SAT-solver,” in *Proceedings of FMCAD*, vol. 1954. Springer LNCS, 2000.